

AD A053048

DDC FILE COPY



12

AD

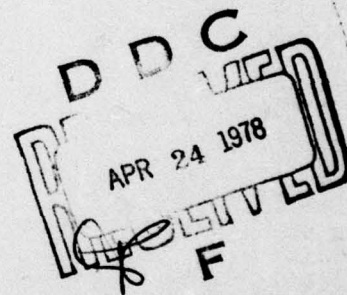
Report 2227

MULTISENSOR TARGET ACQUISITION

MODEL COMPARISON

by
Albert Perri

November 1977



Approved for public release; distribution unlimited.

U.S. ARMY MOBILITY EQUIPMENT
RESEARCH AND DEVELOPMENT COMMAND
FORT BELVOIR, VIRGINIA

Destroy this report when it is no longer needed.
Do not return it to the originator.

The citation in this report of trade names of commercially
available products does not constitute official endorsement
or approval of the use of such products.

14 MERADCOM

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. NUMBER 2227	2. GOVT ACCESSION NO.	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) MULTISENSOR TARGET ACQUISITION MODEL COMPARISON	5. TYPE OF REPORT & PERIOD COVERED Final Report	6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s) Albert Perri	8. CONTRACT OR GRANT NUMBER(s)	
9. PERFORMING ORGANIZATION NAME AND ADDRESS Commander, US Army Mobility Equipment Research and Development Command, ATTN: DRDME-RT Fort Belvoir, Virginia 22060	10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS	
11. CONTROLLING OFFICE NAME AND ADDRESS Director Defense Nuclear Agency Washington, DC 20305	12. REPORT DATE November 1977	13. NUMBER OF PAGES 105
14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office) Director TRADOC Systems Analysis Activity White Sands Missile Range, NM 88002	15. SECURITY CLASS. (of this report) Unclassified	15a. DECLASSIFICATION/DOWNGRADING SCHEDULE
16. DISTRIBUTION STATEMENT (of this Report) Approved for public release; distribution unlimited.		
17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)		
18. SUPPLEMENTARY NOTES		
19. KEY WORDS (Continue on reverse side if necessary and identify by block number) Target-acquisition Models Military Worth Multisensor Sensor Models Combat Simulation Scenario Survivability Computer Model Probability of Detection Vulnerability Intelligence Target-locating Error Theater Nuclear Force Survivability Sensor Flow		
20. ABSTRACT (Continue on reverse side if necessary and identify by block number) The purpose of this effort is to review existing computer models and methodologies and to select one which best satisfies the target-acquisition requirements of the Theater Nuclear Force Survivability (TNF/S) Program. Of the methods and models reviewed, only the following four models appeared to have any possibility of accomplishing the necessary multisensor multitarget tasks: (1) The SCREEN/CRESS Model; (2) The CAMWTH Model; (3) The SAI Combat System Survivability Model; and (4) The STANO-SAM IV Model. <i>→ next page</i>		

(Continued)

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

i

403-160

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE(When Data Entered)

(Block 20 Cont'd)

The report concludes that none of the models as they now stand is adequate for the needs of the TNF/S Program. Each of the models could, however, with adequate modifications serve the TNF/S Study needs.

The report concludes that the CAMWTH model be used as a base for the target acquisition requirements of the Theater Nuclear Force Survivability Program.

This alternative was chosen as that best suiting the needs of the TNF/S Study with regard to what target acquisition data is presently available or could be reasonably made available from planned tests for model input, timeliness of TA model availability, resource restrictions for model modification and testing, and quality of anticipated results.

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE(When Data Entered)

CONTENTS

Section	Title	Page
	ILLUSTRATIONS	iv
I	INTRODUCTION	
	1. Purpose	1
II	INVESTIGATION	
	2. General	2
	3. Models Reviewed	5
	a. The SCREEN/CRESS Model	5
	b. The CAMWTH Model	23
	c. The SAI Combat Survivability Model	36
	d. The STANO-SAM IV Model	46
	4. Overall Model Comparisons	62
	5. Alternatives for the TNF/S Study	65
III	CONCLUSIONS	
	6. Conclusions	72
	APPENDICES	
	A. SCREEN Input Data	73
	B. Physical Data Requirements	74
	C. Types of Aerial and Ground Sensors Modeled	75
	D. Camera System Parameters	76
	E. CAMWTH Inputs	77
	F. CAMWTH Outputs	79
	G. Scenario Data	80
	H. CAMWTH Surveillance Systems Inputs	82
	I. SAI Model Inputs	83
	J. SAI Model Outputs	84
	K. STANO-SAM IV Inputs	85
	L. STANO-SAM IV Outputs	89
	M. PRERUN Inputs and Outputs	89
	N. MSM Outputs	93
	O. Current STANO-SAM Dimension Limits	95
	P. STANO-SAM Sensors	96

ILLUSTRATIONS

Figure	Title	Page
1	A Possible Flight Path	9
2	SCREEN-AIR Computer Model Block Diagram	12
3	SCREEN-GROUND Computer Model Block Diagram	13
4	Control Copy Output	18
5	Intelligence Copy Output	20
6	Structure of the Program and Major Functions of Each Module of the CAMWTH Model	24
7	Job Steps and Data Flow of the STANO-SAM IV Model	50

ACCESSION for ☒ on ☐

NTIS ☐

DDC ☐

UNANNOUNCED ☐

JUST NOTICE ☐

BY ☐ DISTRICT/STATE/ACTIVITY CODES ☐

A ☐ SPECIAL ☐

MULTISENSOR TARGET ACQUISITION

MODEL COMPARISON

I. INTRODUCTION

1. **Purpose.** The purpose of this effort is to review existing computer models and methodologies and to select one which most realistically and best satisfies the target-acquisition requirements of the Theater Nuclear Force Survivability (TNF/S) Program. The goals of the target-acquisition program are twofold. The first goal is to provide direct, target-acquisition input to USA TRADOC Systems Analysis Activity (TRASANA) for use in division-/corps-/theater-level, combat-simulation models. These models are used to perform the overall net assessment of theater nuclear force survivability. The second goal is to conduct an independent unit and intermediate level TNF target-acquisition analysis to determine, prioritize, and quantify sensor threats to theater nuclear forces; to develop a measure of survivability due to changes in this sensor threat; and to recommend fixes in equipment, tactics, and training necessary to increase TNF survivability.

In order to accomplish both goals, an accurate target-acquisition data base which will be capable of inputting combat simulations of all types and force levels (from squad to entire theater) must be established. This data base must be traceable to and be established on sound, experimental results. This differs from present corps- and theater-level, target-acquisition data bases which are usually derived from Delphic determinations. The effort is further complicated by a lack of existing experimental results upon which to base tabulated values.

The computer model or methodology selected by this effort must be capable of creating this target-acquisition data base from either known target (environment) sensor facts or experimental, probability-of-detection data. The output required is the total, target-acquisition probability of detecting different size and complexity targets from the combined effects of all the sensors trying to find those targets.

It is also desired that the selected methodology or model be able to directly assess casualties or to easily input a model that can assess casualties at the unit, company, or division level. This is necessary to evaluate the military effectiveness of proposed, target-acquisition countermeasures.

Of the methods and models reviewed, only four models appeared to have any possibility of accomplishing the necessary tasks. The four candidate models are:

- a. SCREEN/CRESS.
- b. CAMWTH.
- c. SAI Combat Survivability.
- d. STANO-SAM IV.

Each of these models will be described in detail. Assumptions and "ground rules" for this model investigation are specified in section II, paragraph 2. The models will be described individually, an overall comparison of model capabilities will be made, and possible alternatives for the target-acquisition requirements of the TNF/S program will be outlined. The results of this effort will allow wargames to determine the probability of detection for each unique target in the array rather than simply basing detection upon a fixed percentage applied by target type and distance from the forward edge of the battlefield (FEBA) as is now the common practice with percent of knowledge (POK) tables.

II. INVESTIGATION

2. **General.** In undertaking this effort, some assumptions must be made as to what kind of information the models must provide before a comparative analysis can be made.

First, the word, "target," and hence, "target acquisition," can have several different meanings. A target is generally defined as an entity against which ordnance may be directed. Therefore, the definition of target must also depend on the weapon's system. If the weapon is a rifle or gun, a target is a physical object such as a man or a tank. Hence, "target acquisition" is simply the ability to acquire this object. However, if the weapon is an indirect-fire system such as artillery or rocket systems, targets are usually combinations of objects (units). In this situation, "target acquisition" encompasses not only the detection/identification of objects but also the determination of the identity of the units in which the objects may be.

Each specific sensor in the enemy sensor system will have some probability of acquiring each specific target. However, what is generally important in a battle is the total probability that the specific target unit can be acquired by any of the sensor systems in the enemy array which are capable of individually or collectively providing target information to weapon's systems. Hence, a target-acquisition methodology must address the problem of how to combine the detection capabilities of all sensor assets of the enemy surveillance system. Bearing these facts in mind, let us use the following definitions throughout this study:

- a. **target element** — physical objects (e.g., personnel, tanks, trucks) which make up targets.

b. *target* – organization military unit (e.g., 155 SP Howitzer Battery, Tank Company).

c. *one-on-one models* – those models which compute acquisition probabilities of target elements by individual sensors under a specific set of conditions.

d. *target definition* – that process by which detected target elements are combined into detected targets.

e. *one-on-many models* – those models or methodologies capable of performing the target-definition function.

f. *total-target-acquisition probability* – probability that a target will be detected by at least one sensor in the sensor array in a given time frame or as a function of time.

g. *many-on-many models* – Those models or methodologies capable of computing the total-target-acquisition probability.

It is assumed that the model or models to be used by the TNF must be able to supply the probability of detection and identification of target units. Hence, pure one-on-one models will not suffice. It is necessary for the units to be deployed or to be maneuvered in a realistic, tactical scenario. Hence, the TNF target-acquisition model must have the capability of simulating this tactical scenario or at least of being sensitive to changes in the scenario. Similarly, the sensor system must also be simulated or considered in tactical situations. Most of the available data is in the form of one-on-one detection probabilities or sensor-system and target-element characteristics. Therefore, the model should be sensitive to changes in this type of data. This implies that the models should have some form of one-on-one model incorporated in them. The models may be very crude (such as simple, look-up tables), but they should be able to simulate the interaction of the one-on-one detections in a tactical scenario. The models must have some means of performing the "target-definition" function. That is, there must be a means of combining detected-target elements into detected targets or of combining detected element-detection probabilities into target-acquisition probabilities for individual sensors. Finally, the models must have the capability to compute the total, target-acquisition probability or at least to consider the impact of the total, enemy-surveillance system.

Each of the four candidate models will be evaluated on how well it can perform all of the above functions. For each model, the following information will be summarized:

- (1) Overview of the Model.
- (2) Scenario Methodology. (How does the model handle tactical scenario for targets?)
- (3) Surveillance Methodology. (How does the model handle the tactical scenario for the sensor system?)
- (4) One-on-one Models. (What types of sensor are modeled and what is their form?)
- (5) Target Definition. (What procedure is used to combine detected-target elements into detected targets?)
- (6) Total Target Acquisition. (What procedure does the model use in considering the impact of the total surveillance system on detecting target units?)
- (7) Other Model Features. (Does the model have other capabilities other than target acquisition?)
- (8) Summary. (Includes areas which must be changed or augmented if the model is to be used for the TNF study.)

All of the models have strengths and weaknesses in various areas. For example, SCREEN and SAM-STANO have the advantage of having detailed one-on-one models. CAMWTH has look-up tables for one-on-one detections. SAI has no one-on-one considerations.

In addition to the methodologies (or lack of them) in performing the above tasks, consideration must be given to the overall model approach and results. All of these models consider and compute probabilities at some stage of their work. However, two of the models (SCREEN and SAM-STANO) are pure stochastic simulations. The probabilities are played against random-number draws to determine whether or not various events occur at each stage of the simulation. The results of their exercise are, therefore, in terms of probabilities but only as possible outcomes of the exercise. In theory, Monte-Carlo techniques can be used to determine probabilities; but this technique requires many iterations, these models are too expensive to run, and the results are too cumbersome to analyze for many iterations. They can be used with suitable modifications to develop time-ordered target lists. CAMWTH is, essentially, an expected-value model. However, considerable scenario detail is sacrificed in this model over SCREEN or SAM-STANO. In addition, the present output of the CAMWTH target-acquisition module is in the form of target lists for latter sections of

the model. Hence, some modifications would be necessary to compute target-acquisition probabilities. The SAI target-acquisition model does compute and output target-acquisition probabilities and then uses them to create target lists. The model has no one-on-one or even one-on-many methodology in that target-unit-acquisition probabilities for a given sensor must be input. Depending on the depth of analysis required, it may be necessary to use two or more of the above models to accomplish the goals of the TNF target-acquisition study. These alternatives will be outlined in detail.

3. Models Reviewed.

a. The SCREEN/CRESS Model.

(1) **Overview of the Model.** The Stanford Research Institute, Counter-surveillance Reconnaissance Effectiveness Evaluation (SCREEN) Model consists of two sub-models – SCREEN-AIR, for evaluating the effects of airborne surveillance systems; and SCREEN-Ground, for evaluating the effects of ground surveillance systems. Also included in the evaluation is the CRESS-S model for SIGINT systems. Each of the sub-models is independent and, therefore, can be used separately if desired. All models require a large-scale, 65,000-word computer with random-access-disk capability. The program is currently designed for the CDC-6000 series computers.

The capabilities of the SCREEN model allow the user to simulate the time history of the units and sensors in the scenario. This time-ordered simulation allows for target movement and time-varying postures, flight and sensor deployment, and communication delays. The model also simulates the operational degradation of sensor-system performance caused by the atmosphere, equipment failures, attrition, and platform-location error. The mathematical sensor models use physical parameters (sensor performance characteristics and object size, shape, and contrast), environmental parameters (atmosphere and terrain), and operational parameters (range, CS techniques, and deployment) to determine probabilities of detection, recognition, and identification. Once these probabilities are determined, random-number draws based on the probabilities of detection (e.g., the probability of detection is 80%; if the random-number draw is less than or equal to 0.8, the item is detected), recognition, and identification are made which determine exactly what number of objects is then detected, recognized, identified, misrecognized, and misidentified. The model also determines how and when these results are reported to the enemy intelligence organization.

The output of the computer program consists of two portions – control copy and intelligence copy. The control copy provides a complete account of the interactions between sensor systems and the target arrays. It provides not only the

objects which are detected, recognized, and identified but also the associated probabilities and predicted errors. The intelligence output presents, in a time-ordered sequence, only the information the enemy surveillance organization would receive.

The computer is used to store the vast quantity of scenario data; to access this data to determine when a sensor-object interaction occurs; to perform the calculations to determine the results of the interactions; and then to store, order, and output these results in a manner suitable for analysis. Appendix A shows the necessary data which must be input to the model.

In the following description, the numbers (e.g., 1a, 2a, etc) refer to input categories from Appendix A.

When a scenario is written, the position and movements of military units must be plotted on a map grid. This target information (1a) is then taken from the map and is input to the model. In addition to time and position coordinates, target data also includes terrain information, vegetation coverage, background material types, antiaircraft capability of the unit, and number and type of objects present in the target. In this manner, the location in position and time of every object is stored for later use. In addition to this target information, the weather conditions (1b) which prevail during the military operation must be input. Thus, the "scenario history" of the military operation is stored in the memory of the computer.

Once the scenario data is input, the military situation is set. The model then requires physical data to perform the sensor physics calculations. These sensor calculations determine the probabilities of detection (Pd), recognition (Pr), and identification (Pi) for the sensor-object interaction. This physical data consists of object data (2a), material properties (2b), and sensor data (2c). A breakdown of types of object and material data is listed in Appendix B. The sensor performance data depends on the type and quality of the sensor used. Appendix C shows the types of sensors modeled in the SCREEN-AIR and the SCREEN-GROUND. The addition of other sensor types is possible but may require a fair amount of programming even assuming that a usable model exists. This sensor performance data is used by the sensor models to determine how effective a given sensor will be in detecting, recognizing, and identifying objects. (As an example, Appendix D indicates the data necessary for the photographic sensor.)

(2) **Scenario Methodology.** The friendly (BLUE) military scenario can be as complex as one desires. The definition of a "target" in the model is simply a collection of target elements (objects). The SCREEN model calculates sensor interactions at the object level. One or more objects at a given location and time are designated as a target for model purposes. Decoys can be included as some or all of the

target objects in a given target. Targets can be any organizational size or can be individual items if desired. The results of any sensor interaction yield only the number of objects which are detected, recognized, or identified but do not indicate the type of target to which the elements belong. Although the organization size of the model target is completely optional, a lack of realism would most likely occur if the targets are made too large. In the model, a battlefield is subdivided into a grid with 100- by 100-meter cells. These distances are needed for sensor calculation. Therefore, it would be advisable to use targets whose radii were significantly larger than 100 meters. In addition, since sensor calculations are made at the object level, it would be unsatisfactory and certainly inaccurate to simulate that all the target objects in a large unit (say, a battalion) would be collocated at a single point. Therefore, the largest physical unit which normally would be made into a single-model target should be of company size. The unit could be smaller (e.g., firing section) if required.

Different postures and movement may be simulated for each physical target. This is done simply by creating several model targets with incremental displacements along their path of motion and sequential times of existence with different postures. Although there is a limit of 750 model targets for any model execution, this restriction can be circumvented by simply breaking up the scenario into time blocks so that the number of allowable model targets is not exceeded for any model execution.

Background data is supplied as part of the target information. Hence, with the above data, the entire "scenario history" of the targets can be supplied to the model.

SCREEN can be used to simulate almost any size scenario. The limitation is that a great number of man-hours are required to specify the positions, locations, movements, and postures of a large number of targets.

Some feeling for the amount of programming required for scenario writing in SCREEN can be deduced from the following example scenario exercise. A 3-day event for a separate armored brigade was modeled. The physical units in the brigade consisted of the following:

- 3 Armored Battalions.
- 2 Mechanized Infantry Battalions.
- 1 Mechanized Artillery Battalion.
- 1 Administrative Battalion.
- 3 Attached Companies (HQ Company, Armored Cavalry Troop, Engineer Company).

Model units were all of company size with the exception of items which were separated from the parent companies at various times during the scenario (e.g., POL trucks, Commo vans, and elements of the Engineer company). All units in the scenario including various postures and movements required approximately 800 model targets and required approximately 2 man-weeks of coding time to complete. However, a very comprehensive "scenario history" of the military action was made available to the reconnaissance models.

(3) Surveillance Methodology.

(a) Aerial Surveillance (SCREEN-AIR). Once the scenario and physical parameters are set, the model is capable of performing the calculations to determine the outcome of any aerial-surveillance plan devised. The scenario writer must input a flight plan (or group of flight plans), and then the computer will perform the remaining bookkeeping and sensor calculations as necessary.

The man-computer interaction for a typical SCREEN-AIR exercise is described as follows. First, the scenario writer must supply the scenario and physical data previously described. Then, a reconnaissance planner will outline a series of reconnaissance/surveillance (RS) overflights with the number of flight patterns and RS areas completely arbitrary. (An example of a flight pattern is shown in Figure 1.) After this data is input, the computer will perform the calculations as follows.

From the takeoff point, the computer calculates the actual flight path taking into account position errors caused by navigation.

On each leg of an RS area, the swath covered by on-board sensors is calculated. The model then references the scenario history to determine which targets fall within the swath at time of overflight and, therefore, are under the scrutiny of the particular sensor.

Sensors are considered to be turned on only while these legs are flown, during which time the platform is flying straight and level. If a target is covered by a sensor, the interaction between the sensor and all the objects in that target is assumed to occur at the moment of closest approach on that leg. If the sensor has a wide enough swath compared with the leg spacing and target offset, a given target-sensor interaction may occur on more than one leg. Another type of multiple interaction occurs when the aircraft has aboard both a side-looking sensor and a vertical sensor. It is possible that the target will be processed twice — once by the vertical instrument when the target is overflown by the aircraft, and once on the next leg when the side-looking sensor could sense the target even though the target is then outside the maximum, horizontal range of the vertical sensor.

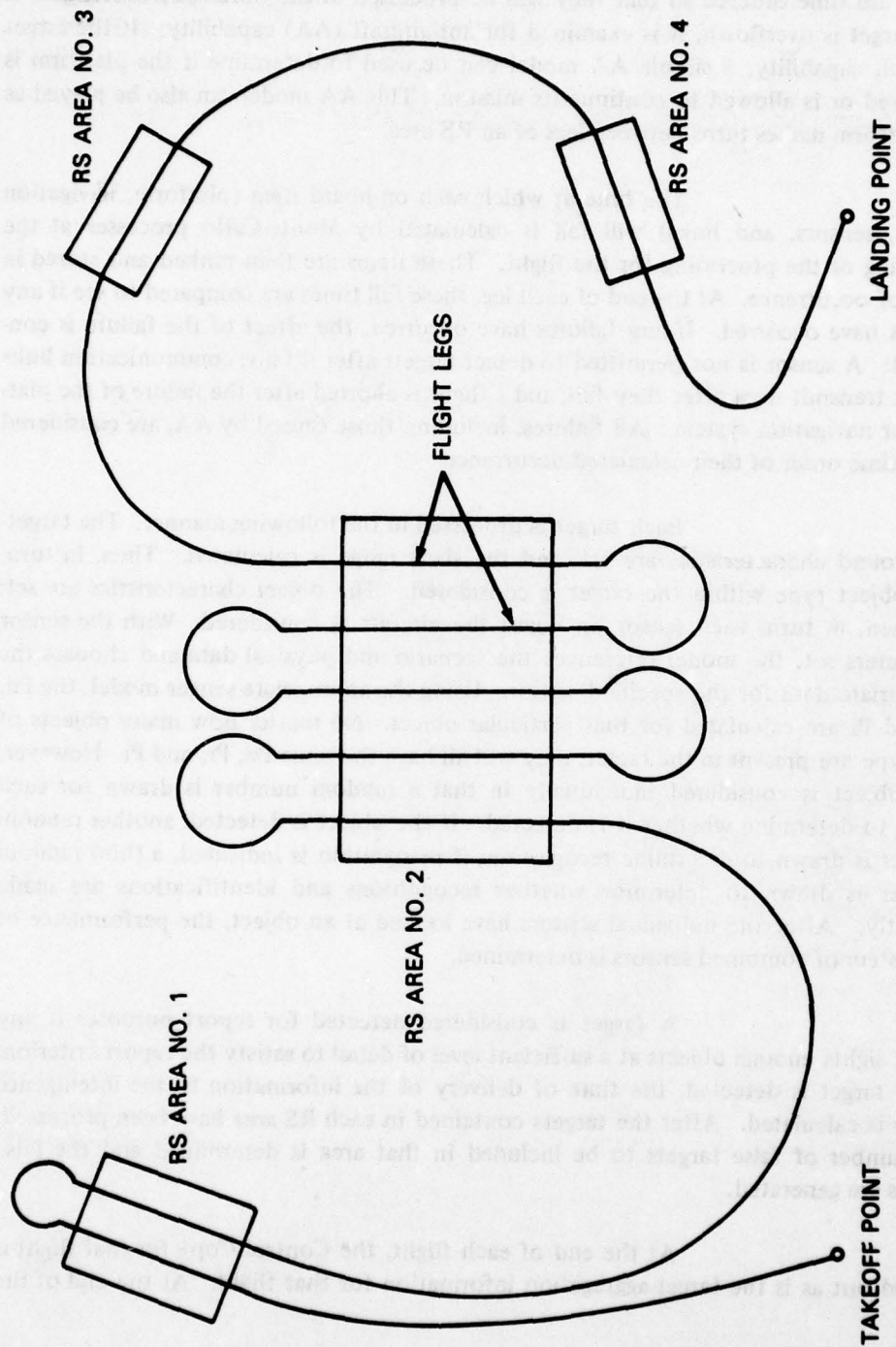


Figure 1. A possible flight path.

The flight time to each target on a leg is calculated, and the targets are time ordered so that they will be processed in the order of overflight. As each target is overflown, it is examined for anti-aircraft (AA) capability. If the target has such capability, a simple AA model can be used to determine if the platform is destroyed or is allowed to continue its mission. This AA model can also be played as the platform makes turns between legs of an RS area.

The time at which each on-board item (platform, navigation system, sensors, and links) will fail is calculated by Monte-Carlo processes at the beginning of the processing for the flight. These items are then ranked and stored in order of occurrence. At the end of each leg, these fail times are compared to see if any failures have occurred. If any failures have occurred, the effect of the failure is considered: A sensor is not permitted to detect targets after it fails; communication links do not transmit data after they fail; and a flight is aborted after the failure of the platform or navigation system. All failures, including those caused by AA, are considered in the time order of their calculated occurrence.

Each target is processed in the following manner. The target-background characteristics are set, and the slant range is calculated. Then, in turn, each object type within the target is considered. The object characteristics are set; and then, in turn, each sensor on board the aircraft is considered. With the sensor parameters set, the model references the scenario and physical data and chooses the appropriate data for the specified sensor. Using the appropriate sensor model, the Pd, Pr, and Pi are calculated for that particular object. No matter how many objects of that type are present in the target, they will all have the same Pd, Pr, and Pi. However, each object is considered individually in that a random number is drawn for each object to determine whether it is detected. If the object is detected, another random number is drawn to determine recognition; if recognition is indicated, a third random number is drawn to determine whether recognitions and identifications are made correctly. After the individual sensors have looked at an object, the performance of the system of combined sensors is determined.

A target is considered detected for report purposes if any sensor sights enough objects at a sufficient level of detail to satisfy the report criterion. If the target is detected, the time of delivery of the information to the intelligence center is calculated. After the targets contained in each RS area have been processed, the number of false targets to be included in that area is determined and the false targets are generated.

At the end of each flight, the Control Copy for that flight is printed out as is the target-aggregation information for that flight. At the end of the

last flight, all the target reports are ordered in terms of their arrival times to the intelligence team, and the Intelligence Copy is printed out in that order.

A block diagram for the general logic flow of the SCREEN-AIR computer model is shown in Figure 2.

(b) Ground Surveillance (SCREEN-GROUND). SCREEN-GROUND is similar to SCREEN-AIR in that it requires the same scenario information and data format. The sensors require different technical data depending on sensor type. SCREEN-GROUND simulates the intelligence (or targeting) data which would be obtained from an observation post (OP) or a reconnaissance patrol (Patrol).

The man-computer interaction for ground reconnaissance is less automated. Again, the scenario writer must supply the scenarios and technical data. The reconnaissance planner must then specify time and positions of all reconnaissance patrols. However, unlike SCREEN-AIR which handles all the bookkeeping concerning whether or not targets are under the purview of the sensors, many of these details must be handled manually for SCREEN-GROUND. The Line-of-Sight (LOS) between an OP and an appropriate model target must be specified. For Patrols, a sequence of OPs must be created along the patrol route specifying both LOS and sighting time for each applicable target.

Although this data is not necessary for targets not in LOS to particular OPs and Patrols, this task can be extremely tedious and time-consuming for large scenarios. If the program is to be used with large scenarios, it is absolutely necessary to devise a pre-processor to generate this information automatically. The pre-processor would not be difficult to create if one is willing to accept LOS probabilities based only on OP-target distance and height relationships. However, it would be a difficult task to incorporate a deterministic LOS based on contour and vegetation data for a real battlefield. This would also require additional large data storage requirements.

Sensor-object processing (including OP/Patrol attrition and equipment failures) for SCREEN-GROUND is similar to that for SCREEN-AIR. A control copy of each OP/Patrol is printed out as it is processed. At the end of all OPs and Patrols, all sensor reports are time ordered in terms of their physical arrival times to the intelligence team, and the Intelligence Copy is printed out.

A block diagram for the general logic flow of the SCREEN-GROUND program is shown in Figure 3.

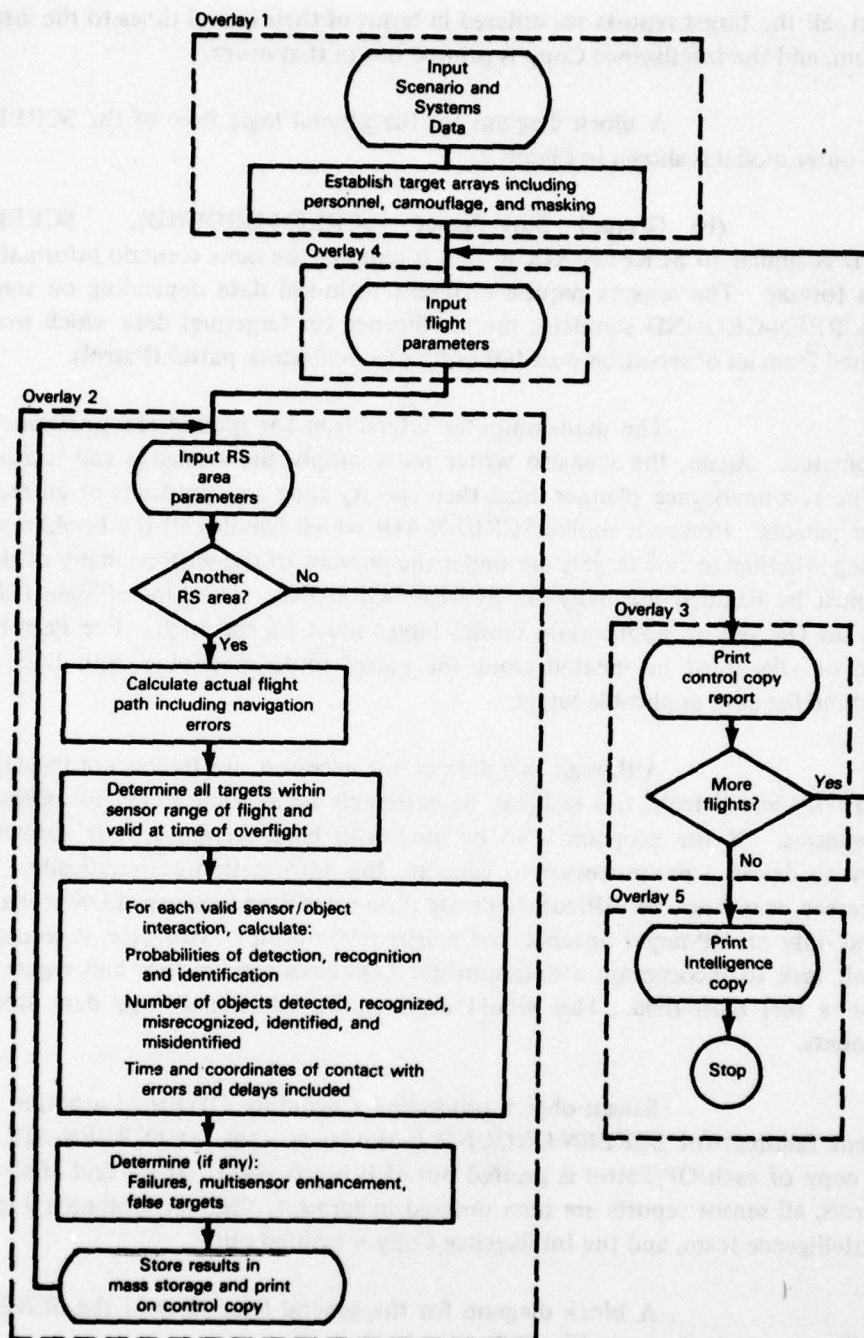


Figure 2. SCREEN-AIR computer model block diagram.

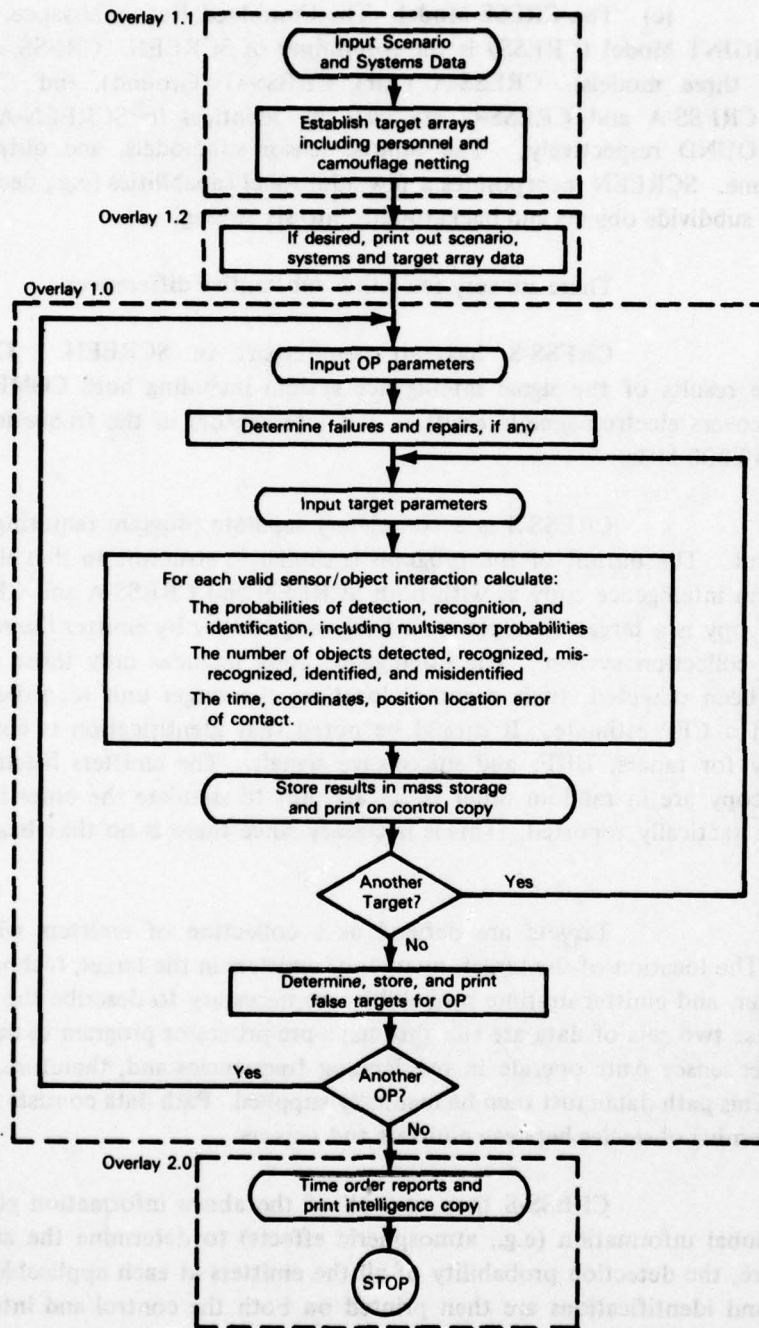


Figure 3. SCREEN-GROUND computer model block diagram.

(c) **The CRESS Model.** The Combined Reconnaissance, Surveillance, and SIGINT Model (CRESS) is the forerunner of SCREEN. CRESS was subdivided into three models: CRESS-A (Air), CRESS-G (Ground), and CRESS-S (SIGINT). CRESS-A and CRESS-G are virtually identical to SCREEN-AIR and SCREEN-GROUND respectively. The inputs, sensor submodels, and outputs are nearly the same. SCREEN incorporates a few additional capabilities (e.g., decoys and the ability to subdivide objects and backgrounds into six subregions).

There are very few other substantive differences.

CRESS-S has no counterpart in SCREEN. CRESS-S simulates the results of the signal intelligence system including both COMINT and ELINT. It covers electromagnetic emitters and interceptors in the frequency range from 0.1 to 40,000 MHz.

CRESS-S is a completely separate program requiring totally different input. The output of the program is similar in structure in that there is a control and an intelligence copy as with both SCREEN and CRESS-A and CRESS-G. The control copy is a target-by-target record showing emitter by emitter the results of the SIGINT collection system. The intelligence copy includes only those emitters which have been detected, their reported location, the target unit identification if possible, and a CEP estimate. It should be noted that identification is considered possible only for radars, UHF, and microwave signals. The emitters listed on the intelligence copy are in random order in an attempt to simulate the order in which they may be tactically reported. This is necessary since there is no time base in the model.

Targets are defined as a collection of emitters which are collocated. The location of the target, number of emitters in the target, technical data on the emitter, and emitter up-time probability are necessary to describe the SIGINT system. These two sets of data are run through a pre-processor program to determine which emitter-sensor pairs operate in overlapping frequencies and, therefore, require path data. This path data must then be manually supplied. Path data consists of information concerning obstacles between emitters and sensors.

CRESS-S then uses all of the above information plus some additional global information (e.g., atmospheric effects) to determine the audibility and, therefore, the detection probability of all the emitters at each applicable sensor. Detections and identifications are then printed on both the control and intelligence copies. Since there is no time base, CRESS-S has no capability for analyzing changing military scenarios except by using a series of "snapshot" situations with separate scenarios and executions of the program.

Due to the above limitations, CRESS-S can be extremely tedious to use for large-scale simulations unless it is revised and some method of inserting a time base into the simulation is created.

(4) One-on-one Sensor Models in SCREEN. The one-on-one sensor submodels are the core of the SCREEN models. In general, these submodels are rather old sensor subroutines. The majority of the submodels were taken directly from the sensor submodels in the CRESS computer program with very few modifications. Most of these models were developed or adapted by the Stanford Research Institute (SRI) for the U.S. Army "TARS-75" study in 1966 and the U.S. Marine Corps study "Systems Analysis of Advanced Target Acquisition Systems (U)" in 1967. The thermal, TV, Laser, Camera, and Passive Night Vision Device Models were developed by SRI in conjunction with Honeywell, Inc., for the above two studies. The SLAR radar model is a direct adaptation of a Honeywell model developed during the study "Mathematical Model Reconnaissance and Penetration Study" in 1965. The visual model is an adaptation of the Franklin and Whittenberg Air-to-Ground Model of 1965. The Ground Surveillance Radar Model is taken from the Honeywell Study "Ground Sensor Methodology" in 1966.

The sensor submodels all assume that the target object is completely described by the following parameters:

- Dimensions (length, width, and height).
- Average reflections in the appropriate sensor bands (visual, near IR, Radar cross-section).
- Object temperature.
- IR emissivity.
- Pattern of reflectivity/emissivity. (It is possible to subdivide objects into up to six different regions.)

The background is represented by similar parameters.

All of the target and background information is supplied as part of the scenario data. In addition to these data, other environmental and operational data are either supplied as part of a model or computed by model bookkeeping as the scenario progresses.

The sensor subroutines use this data to do the technical sensor physics calculations to compute the probabilities of detection, recognition, identification, misrecognition, and misidentification of each object which the sensor covers. All objects in the same target will have the same probabilities. A decision as to

whether or not a given object is required (and to what level of discrimination) is determined by random-number draws against these probabilities.

In general, the sensor submodels are not very detailed. That is, there are probably better (certainly, more current) one-on-one sensor models available. However, these models are sufficient for the purpose of a large, war-game simulation. Although the detail in the models is probably insufficient for predicting the technical effectiveness of specific camouflage techniques, the models do consider the technical details in the required depth and many of the operational variables for the simulation of large-scale reconnaissance endeavors. Many of these variables (i.e., vegetation masking, intentional background blending, directed search enhancement, camouflage netting, sensor on-off times, and decoy effectiveness) are under the control of the scenario writer. However, once the scenario is fixed, many other operational variables such as object-to-sensor range, time under sensor view, terrain masking, sensor failures, proximity enhancement, platform failures, and shadows are handled by model book-keeping. The fact that the sensor-object interactions are computed individually also adds to the realism of the simulation.

An additional capability of SCREEN one-on-one submodels is multispectral enhancement. This capability simulates the synergistic effects of two or more sensors of different types. That is, two different sensor types on the same platform/OP would have a higher probability of acquisition than the combined effects of both sensors looking independently. The synergistic increase would depend on sensor types and object classes. This effect is treated as a separate sensor called "MULTI."

This sensor submodel was devised by SRI based on research conducted by the Honeywell Study, "Advanced Surveillance Systems Investigation Through Simulation on TARS (ASSIST)" (U), in 1967.

While the list of sensor types is not all-inclusive, the modular design of SCREEN allows the insertion of other sensor types if they can be accommodated by existing platforms (aircraft, OPs, and Patrols).

With the exception of the visual model, the sensor submodels have not been compared with results of field experiments. The visual submodel owes its basic formulation to field data from the Franklin-Whittenberg test. However, the model is very empirical, and the field-test conditions were somewhat limited. This lack of experimental verification is not believed to be serious. True acquisition data generally exhibits greater dependence on operational and situational variables than it does on technical variables. The sensor submodels in SCREEN can be manipulated by the scenario writer to insure adequate agreement with any reasonable set of field

data with reasonable technical input. The technical details are modeled in sufficient depth to maintain sensitivity to changes in these parameters which may occur in a large exercise.

In summary, the one-on-one models in SCREEN may be simplistic in their technical depth, but they are likely sufficient for large reconnaissance-surveillance exercises. The SCREEN model possesses attractive options for use of the one-on-one models in a large-scale simulation.

(5) Target Definition and Total-Target Detection.

(a) Model Outputs. The outputs of both models (air and ground) are similar and contain two parts – control copy and intelligence copy. Both reports are made at the object level. The control copy can supply all of the information concerning the one-on-one, object-sensor interactions. In this sense, SCREEN can be used as a one-on-one model. The intelligence copy supplies only raw detections, recognitions, and identifications which the enemy surveillance system would obtain. (NOTE: This may include false targets, misrecognitions, and misidentifications.) The model will also compute summaries of the number and ratios of detected/undetected objects in various categories.

1 Control Copy. During program execution, a complete history of all sensor-target interactions is printed out on the control copy. The control copy contains:

- Flight and equipment information (SCREEN-AIR).
- OP/Patrol information (SCREEN-GROUND).
- Probability of detection, recognition, and identification for each object in all targets covered by any sensor.
- Number of objects detected, recognized, and identified.
- Any misidentified or misrecognized objects.
- False targets.
- True target locations.
- Sensor contact time.
- Report time.
- Reported position.

Figure 4 is an example of a control copy report for SCREEN-AIR.

2 Intelligence Copy. The intelligence copy presents only that subset of information that the enemy would normally obtain using his sensor

FLIGHT PLATFORM SENSORS TAKEOFF TIME	Flight 1 87TR VIS AS18 G102 111100	SITUATION NAVIGATION FLIGHT TIME	WAR GAME ALTITUDE 2000 SPEED 500 NBR TGTS 19
RECONNAISSANCE/SURVEILLANCE AREA 1	ALTITUDE 2000	SPEED 200	
TIME 111103	TARGET 2012	TRUE LOCATION DT 161878	REPORT GROUP 5 DISTANCE 517
		MII PDM 0 MD 0	
NBR OBJECT	87TR	PLOS=75	AS18 PLOS=87
	R,D PD PR PI OD OB OI PD PR PI OD OR OI PD PR PI OD OR OI		
3 2 1/2-Ton Truck	17,5 72 90 90 2 1 1 17 99 99 1 1 1 84 0 0 3 0 0 72 0 0 2 0 0 78 60 62 3 1 1		
1 1/4-Ton Truck	22,6 -1 0 0 0 0 -1 1 18 99 99 1 1 1 84 0 0 1 0 0 72 0 0 1 0 0 81 0 0 1 0 0		
1 Med Net	10,4 72 90 90 1 1 1 18 99 99 1 1 1 -1 0 0 0 0 0 -1 0 0 0 0 0 0 0 0 0 0 0		
10 Standing Man	30,7 72 5 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 71 0 0 7 0 0 63 4 0 10 0 0		
8 Small Net	10,4 72 90 90 6 6 6 9 99 99 1 1 1 -1 0 0 0 0 0 -1 0 0 0 0 0 0 0 0 0 0 0		
Large Net			
Report Time	111103	111103	111103
Reported Position	UT 160878	UT 160878	UT 160878
CEP-- A, B, C,	0 7 14 28	0 7 14 28	0 7 14 28
TIME 111103	TARGET 1C12	TRUE LOCATION DT 154874	REPORT GROUP 6 DISTANCE 139
		MII PDM 0 MD 0	
NBR OBJECT	87TR	PLOS=75	AS18 PLOS=87
	R,D PD PR PI OD OB OI PD PR PI OD OR OI PD PR PI OD OR OI		
3 2 1/2-Ton Truck	17,5 72 90 90 2 2 2 19 99 99 0 0 0 84 1 0 3 0 0 72 0 0 2 0 0 78 67 62 3 2 2		
1 1/4-Ton Truck	22,6 -1 0 0 0 0 2 -1 1 20 99 99 1 1 1 84 0 0 1 0 0 72 0 0 1 0 0 81 0 0 1 0 0		
1 Med Net	10,4 72 90 90 1 1 1 20 99 99 1 1 1 -1 0 0 0 0 0 -1 0 0 0 0 0 0 0 0 0 0 0		
10 Standing Man	30,7 72 6 0 1 0 0 0 0 0 0 0 0 84 0 0 9 0 0 0 -8 0 0 1 0 0 80 5 0 12 0 0		
8 Small Net	10,4 72 90 90 7 7 7 10 99 99 0 0 0 -1 0 0 0 0 0 -1 0 0 0 0 0 0 0 0 0 0 0		
Med Net			
Report Time	111103	111103	111103
Reported Position	UT 154874	UT 154874	UT 154874
CEP-- A, B, C	0 7 14 28	0 7 14 28	0 7 14 28

Figure 4. Control copy output.

systems. The list contains all reported objects at their calculated highest level of discrimination (i.e., either detected, recognized, or identified). There is no indication as to whether they are real or false targets or whether they have been recognized or identified correctly. The object list is time ordered as to when the raw data will be received by the enemy. Figure 5 is an example of a SCREEN-AIR intelligence report.

One should note that the intelligence copy is again at the object level. There is absolutely no attempt to infer the type of target unit in which the objects are located. There are options available in the model to assist in the use of the intelligence output. These include clustering of objects within certain radii as an aid in determining which reported target objects might be combined to form part of a larger unit. Also, there are different report criteria available which would limit the number of extraneous objects which must be categorized. These reporting criteria can include lists of important object types to be reported and minimum-number thresholds for the reporting of other objects.

(b) Use of the Model to Perform Target Definition and Total-Target Detection. By using only the one-on-one capability, the control-copy output can be used in preparing probability-of-detection tables for various object-sensor interactions. However, the primary purpose for which the model was devised is for SCREEN to be used in a closed war game. For SCREEN to be used in this manner, it is necessary for two or three "teams" to be formed. First, there must be a "control team" to format data and to exercise the model. The control team would form the interface between the model and the other teams. Second, there should be a "scenario team." The scenario team would be responsible for establishing the tactical scenario. If an agreed upon scenario is available and the scenario is unchanging during the model exercise, then the job of this team is merely formatting the data for use by the model. This could be adequately handled by the control team. The third team is the "intelligence team." The intelligence team would be responsible for combining the object reports from the model intelligence output into the intelligence data which the enemy would receive. This, of course, would vary as to what kind of information is being sought and the rules of the war game. The intelligence information could consist of a time-ordered target list for enemy firepower or could consist of the simulated report of an enemy G-2 or S-2 operation (order of battle information). Depending on the rules of the game, it may be necessary for this control team to supply only a limited summary of model output to the intelligence team. It is also possible to allow the intelligence team to schedule subsequent surveillance flights, OPs, or Patrols. In this sense, the control team would act as an interface between two competing teams and the model.

Whatever the rules or procedures used by the intelligence team, SCREEN users must perform the target definition and total-target-detection

RPT TIME 50642 RPT GRP POSIT AE 688041 CEPS IDENTIFIED 0	77 IMG TIME 41846 FLT DC-10 0 7 14 28 RS 1 ALT 1000 SPEED 150 RECOGNIZED 0	AC OV-10 NAV VIS MD 0 RBF 0 SENSOR AFR1 DETECTED 1 VERY SMALL LAND OB	DIST 600
RPT TIME 50710 RPT GRP POSIT AB 495817 CEPS IDENTIFIED 1 ROCKET LAUNCHER	62 IMG TIME 41816 FLT DC-9 0 7 14 28 RS 1 ALT 1000 SPEED 150 RECOGNIZED 0	AC OV-10 NAV VIS MD 0 RBF 0 SENSOR AFR1 DETECTED 1 VERY SMALL LAND OB	DIST 500
RPT TIME 50710 RPT GRP POSIT AB 495817 CEPS IDENTIFIED 1 ROCKET LAUNCHER	62 IMG TIME 41816 FLT DC-9 0 7 14 28 RS 1 ALT 1000 SPEED 150 RECOGNIZED 0	AC OV-10 NAV VIS MD 0 RBF 0 SENSOR MULT VIS AFR1 DETECTED 1 VERY SMALL LAND OB	DIST 500
RPT TIME 50714 RPT GRP POSIT AB 210519 CEPS IDENTIFIED 0	40 IMG TIME 41922 FLT DC-7 0 7 14 28 RS 1 ALT 1000 SPEED 150 RECOGNIZED 0	AC OV-10 NAV VIS MD 0 RBF 0 SENSOR AFR1 DETECTED 4 VERY SMALL LAND OB	DIST 700
RPT TIME 50714 RPT GRP POSIT AB 210519 CEPS IDENTIFIED 0	40 IMG TIME 41922 FLT DC-7 0 7 14 28 RS 1 ALT 1000 SPEED 150 RECOGNIZED 0	AC OV-10 NAV VIS MD 0 RBF 0 SENSOR MULT VIS AFR1 DETECTED 4 VERY SMALL LAND OB	DIST 700
RPT TIME 50715 RPT GRP POSIT AB 545992 CEPS IDENTIFIED 1 RR ENGINE 2 RR BOXCAR 1 RR CABOOSE	74 IMG TIME 41822 FLT DC-10 0 7 14 28 RS 1 ALT 1000 SPEED 150 RECOGNIZED 1 RAILROAD CAR 0 0	AC OV-10 NAV VIS MD 0 RBF 0 SENSOR AFR1 DETECTED 2 LINEAR TARGET 0 0	DIST 200

Figure 5. Intelligence copy output.

functions external to the model. This human analysis of simulated reconnaissance and surveillance data is the most difficult and crucial step for users of the model. No attempt has been made to devise any additional aids for the intelligence team beyond those already mentioned in the previous section. It was the philosophy of the developers of the model that it would be undesirable to take human participation out of this difficult process. Nevertheless, it is desirable and necessary for all users of the model to devise some scheme and some manual or computer-assisted aids for the intelligence analysts. These aids could be very simple summary forms with a straightforward procedure for their use. If the model is to be useful for the TNF study, the intelligence procedure will have to be specified in detail and suitable aids devised.

It may be possible to use the methodologies of one of the other models with suitable revisions. Otherwise, this model would be virtually useless for the TNF study since it has no method of performing the crucial target definition and the total-target-detection functions.

(6) **Other SCREEN/CRESS Features.** SCREEN/CRESS has no features other than the simulation of a reconnaissance system.

(7) **SCREEN/CRESS Summary.** The SCREEN model is a stochastic simulation of the surveillance system. To use the model in the manner for which it was designed requires a man (or men) in the analysis loop. Human analysts (the intelligence team) use model results in terms of detected elements to perform the required target definition and total-target-detection function. These analysts must decide whether or not the acquisition of a set of detected-target elements constitutes a legitimate target, identify the target if possible, and specify the target unit's size. The acquired elements can have three levels of discrimination — detection, recognition, or identification. The analysts must decide whether the predicted level of discrimination is sufficient for target-definition purposes. After doing all these things, the intelligence team can arrive at a time-ordered, detected-target list. There does not appear to be any realistic method of assigning probability numbers to these reports.

The use of human analysts does have some advantages. First, the procedure is probably more realistic. More importantly, the use of human analysts in the exercise enables the simulation of another aspect of the surveillance/intelligence system (i.e., other than detecting targets for hostile firepower). Human analysts would be able to determine order-of-battle information which includes force deployments, disposition, capabilities, and intent. This process requires human interpretation of not only detected elements but also the indicators which confirm or deny the essential elements of information of combat intelligence. This kind of information is often important for the allocation of assets such as targeting priorities.

Even though an intelligence team is used to provide the important conclusions of the model exercise, some modifications as follows are required for the base model:

(a) One-on-one models for other possible sensor systems must be incorporated where they are lacking (e.g., sound ranging).

(b) The LOS probabilities for patrols and observation posts should be computer generated. This should be done probabilistically based on sensor-to-target distance and height differences and could be done either by the model or by a pre-processor. The present requirement of user input for each target - OP/Patrol pair is completely unsatisfactory.

(c) The Electronic Warfare (EW) sensor's model (CRESS-S) must be radically revised so that it can use the same scenario data as the ground and air programs. Also, a time base must be inserted into this model.

(d) A computer sort should be done so that all sensor reports for a given target are listed together in a time-ordered manner.

(e) The report criteria should be used to limit the amount of information that the analysts need consider. Detailed ground rules for making the target lists must be carefully planned in advance. Summary forms outlining the agreed upon target definition procedure could speed the work.

There are some obvious problems in using a man-in-the-loop approach. First, the procedure can be tedious and slow. Even more importantly, lacking a very rigid formal procedure, the information derived is very dependent on the individuals performing the task. These problems should be considered and proper caution exercised before embarking on any SCREEN-like man-in-the-loop analysis.

There is, perhaps, a possible alternative to a man-in-the-loop. One could modify the program to perform a target-by-target sort on the control copy (which contains probabilities) and use this information as input to an algorithm which would use a rigid-target-definition methodology. A modified version of the methodology used in the target-definition section of CAMWTH could be made appropriate.

Even with the limitations and the needed modifications, SCREEN does have some very attractive capabilities:

- A very flexible and realistic scenario capability for both the BLUE target array and the RED sensor array.

- A set of one-on-one sensor models with sufficient technical detail and adequate consideration of important operational variables.
- Time ordering of intelligence output considering processing delays.
- Three levels of discrimination for target-element acquisition with associated probabilities.

The key drawback in using SCREEN is the large amount of effort which must be spent in planning and modifications before the analysis can begin. However, if a high degree of detail is required in the scenario and *technical* one-on-one models are desired, then SCREEN is probably the preferred model. It can be used to supply element detections for a man-in-the-loop target definition procedure or as input to a computerized, target-definition methodology. Either approach could be used to generate a detected-target list.

b. The CAMWTH Model.

(1) **Overview of the Model.** The CAMWTH Model is a simulation of the processes and results of the engagement of friendly forces by enemy, indirect-fire weapon systems. The model attempts to simulate the following four stages of such an engagement:

- Target Detection and Definition (Program MOD1).
- Target Analysis (Programs MOD2C and MOD2N for conventional and nuclear ordnance respectively).
- Fire Planning (Program MOD3).
- Fire-Plan Execution (Programs MOD4C and MOD4N for conventional and nuclear ordnance).

The model is written in FORTRAN IV with an overlay structure suitable for CDC-6000 series machines. Each of the four stages of indirect-fire combat is simulated on separate primary overlays. The structure of the program and major functions of each module are shown in Figure 6. It should be noted that overlay 2 has separate modules for conventional and nuclear ordnance (MOD2C for conventional and MOD2N for nuclear). Similarly, overlay 4 has separate modules for conventional and nuclear fire-plan execution. Each of the separate modules can be exercised independently. Overlays 2 through 4 require data generated by previous overlays. However, this data is transmitted through disk files so it is possible to exercise subsequent overlays during completely separate executions. MOD1 can be exercised independently with only negligible modification.

The overall model is a combination of an expected-value model and a war game simulation. In Program MOD1, the target-element-detection calculations

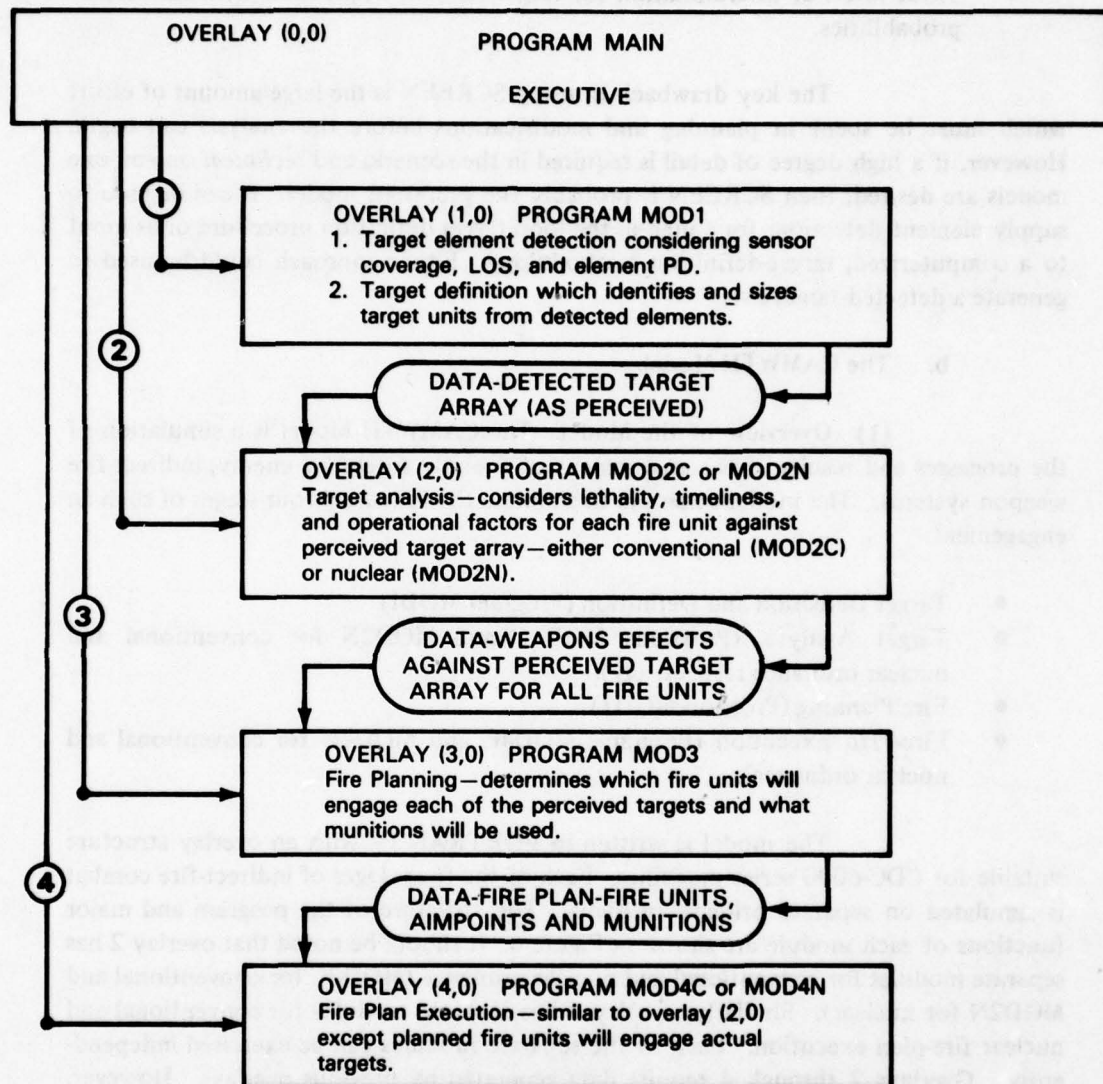


Figure 6. Structure of the program and major functions of each module of the CAMWTH model.

are purely expected value. That is, the number of detected elements per target is simply the number available multiplied by the associated probability. However, the scenario and sensors are input in map-grid coordinates as they would be in a war game. The probability of a target coming under the scrutiny of sensor systems and target-sensor ranges depends critically on target and sensor emplacement on the map grid. Hence, the number of elements detected is very scenario dependent. Target definition is, essentially, expected value except that it depends critically on the number of target elements detected which in turn is very scenario dependent. Damage calculations are purely expected value; however, the fire-planning stage depends on the input scenario.

Each overlay has distinct card input requirements and produces intermediate output for analysis. Appendix E illustrates the input requirements for each module. Appendix F illustrates module output. Greater detail on the input requirements, methodology, and outputs of MOD1 (Target Acquisition Module) will be described in the following sections.

(2) Scenario Methodology. The friendly military scenario is simply an array of up to 500 stationary-target units. The target units are, in turn, collections of up to 4 different types of collocated target elements. The user specifies a target by its type, position, times of existence, composition (target elements), and environment (one of three possible). The targets are generally TO&E military units (e.g., 155 battery or armor platoon), but the user has complete freedom in the specification of the target. He can specify any element. There are 11 different types of target elements available to the scenario. The target-element types are arbitrary (with a few restrictions) and specified by the user.

In addition to scenario targets, the model requires a set of 25 stylized units, each with three different organizational sized (75 total) units. These stylized units are critical to the target-definition portion of the acquisition model and later target analysis. Unit identities determined in the target-definition section will always be one of the stylized units. The first 15 stylized types represent the types of military units used as target units in the scenario. Each scenario target must correspond to one of these 15 stylized-unit types. Although advisable, it is not necessary that all scenario targets have the same composition as the corresponding stylized-unit type and size. The remaining 10 stylized units represent general types of units which the target-definition section will use to classify targets which have been detected by sensors incapable of completely identifying target elements. The model user has nearly complete freedom in assigning target types and compositions for the first 15 stylized units and the composition of the last 10 general, stylized units.

With proper use of time slices, the 500-target limit can be circumvented. Hence, the model can be used in a very large scale simulation such as

brigade-, division-, or corps-size exercises. It is estimated that a division-size exercise could be coded by experienced personnel in 2 man-weeks if the scenario data were readily available.

An overview of constraints and kinds of data required for scenario specification is shown in Appendix G. Also listed are examples for the data.

(3) Surveillance Methodology. The CAMWTH program can model 15 different (user specified) sensor types. The enemy surveillance system is specified by the map-grid deployment and times of existence of up to 500 individual sensors. Each individual sensor must be one of the 15 modeled types, and each sensor of the same type has the same characteristics.

There are three general sensor field of view (FOV) patterns allowed by the model. Each sensor type uses one and only one of them. The first FOV is a circular arc from a fixed-sensor location. This FOV is characteristic of fixed, ground-based sensors such as forward observers, counter battery radar, or ground surveillance radar. Target-to-sensor range is simply the distance from the target to the vertex of the FOV arc. The second general FOV modeled is a fixed, rectangular area. This FOV is characteristic of fixed, ground-based sensors which have a baseline with two or more subsensors distributed along the baseline. Sound ranging, flash ranging, and ground-based direction finding are examples of sensors with this kind of FOV. The target-to-sensor range is the perpendicular distance from the target to the baseline. The third FOV is for moving sensors. The sensor swath is the rectangular areas determined by offsets from sensor line of motion, and the target-to-sensor range is the perpendicular distance from target-to-sensor path. Airborne reconnaissance, patrols, and airborne observation posts are examples of this type of sensor.

In addition to FOV, all sensors of a given type have common target-element-detection probabilities and CEPs (both as functions of range and conditions). Various sensor types depend on an individual target's probability of shooting, moving, or transmitting during the target's time of existence. Most sensors require line-of-sight (LOS) to target. LOS is treated probabilistically as a function of height and range. These probabilities are all input.

Each of the individual sensors in the surveillance scenario is specified by its location (single fixed coordinates, baseline, or path of motion), time of existence, height, and delay time. These sensor deployments and times will have critical influence on the predicted-target list.

Target-sensor processing is done by cycling all scenario targets against each individual sensor. If a target-sensor pair has an overlap in time and FOV,

target-element-detection calculations are made. There is no consideration of time under sensor scrutiny in the detection calculations; and, if detection occurs, the detection time is determined to be the first instant of target-sensor overlap for fixed sensors or the moment of closest approach for moving sensors. If two or more sensors detect a target, the information is considered separately. Therefore, there is no communication or synergism between either individual sensors or different sensor types. These influences can only be considered external to the model (i.e., by creating some artificial "combined" sensor type or by additional model executions with enhanced probabilities).

Sensors cannot detect false-target elements or misidentify real elements. In this sense, the input-element detection is the probability that a sensor detects and identifies the element. The ability of some sensor to identify elements may be restricted from input. For example, ground surveillance radar may only be given the capability of differentiating tracked vehicles, wheeled vehicles, and personnel. It would then be unable to separate APC from tank. It would, however, never misidentify a tank as a wheeled vehicle.

However, the use of decoy-target elements is allowed. When the sensor detects a decoy element, it is determined to be a real element. The possibility of a sensor detecting a decoy and identifying it as a decoy is not explicitly modeled. This possibility must be considered in assigning decoy-detection probabilities. For example, the input probability of detection of a decoy tank is in reality the probability that the item is detected and identified (to the highest discrimination level allowed for the sensor type) as a tank. Hence, the probability that a detection occurs and that the item is identified as a decoy is included in the probability that the object is not detected at all. Target-element-detection probabilities are further discussed in the following section.

Once a sensor detects target elements, the target-definition methodology is invoked and the expected identity, size, and detection probability of the target unit are determined. The target is then included in the detected-target list for later target analysis. This methodology is described in the target-definition section.

Appendix H illustrates the surveillance system inputs and examples of types of sensors to be modeled.

(4) One-on-one Methodology. Target-element-detection probabilities are all inputs to the model. There is no attempt to compute these probabilities on the basis of any physical inputs. All probabilities are range dependent and must be specified for 20 range intervals. There are probability curves for each (target element)-(sensor type) pair under each of six conditions – one for each of three possible target

environments under both day and night conditions. This implies a total of 990 curves of 20 points each. There is also an equivalent set of data for decoy-target elements. This information must reside on random-access disks prior to program execution. There is a small, stand-alone program which reads card input and generates these disk files. The main program also considers a set of fractional degradations to these curves for each of the six conditions of each (element or decoy element)-(sensor type) pair. This input is read during program execution and can be used to selectively degrade these curves to test the effects of countermeasures on the battle outcome.

The number of target elements detected is purely an expected-value computation. These calculations are made for each applicable target-sensor pair in the scenario. Each element in the target is considered separately, and the expected number of detections for each element composing the target is computed. If decoy elements are present in the target, a weighted-average-detection probability is used for the element-detection probability. The target's detection probability is determined by the target-definition routines based on the expected number of detected elements.

This approach for element detection is simple and straightforward. The different target environments can be used to simulate various postures, camouflage conditions, or deployment situations. This allows a great amount of flexibility for the model user. One of the main drawbacks is that of supplying so much data in terms of probability curves.

Another drawback is the way time is handled in the element detections. Time is not explicitly considered in this calculation. Currently, there is an adjustment made in the model for the maximum number of elements a given sensor can detect in a given period, but there is no consideration of the detection probability as a function of sensor viewing time. This approach is perhaps correct for snapshot-type sensors such as airborne photography but is inadequate for obviously time-dependent sensors such as sound ranging or radio direction finding. For these sensors, time considerations must be made external to the model and used to affect the detection curves. This inhibits the model's use in scenarios where target movement must be modeled.

The modeling of time-dependent sensors can probably be simply overcome if time-dependent probabilities are assigned to targets and sensors (i.e., target probability of shooting per unit time and sensor probability of intercept per shot). There appears to be no way of reducing the amount of detection if one desires to maintain range and condition dependence for an effective number of target-element and sensor types.

(5) **Target Definition.** Of the four models under consideration, CAMWTH is unique in that it has a formal, target-definition methodology. On the basis of the expected number of detected elements, the single-element-detection probabilities, the input stylized units, and input minimums for detected targets, this methodology computes the expected unit identity and size as perceived by the enemy. It also determines the probability that a given sensor will detect the target unit and the expected location error associated with the detection. This information is used to create a detected-target list to be used by later target analysis and damage calculations. The methodology for creating a detected-target list will be further described in the following section.

The expected unit identity is determined from an input-identity matrix. This matrix specifies the unit type (one of the stylized-unit types) based on the presence of various combinations of detected-target-element types. The expected number of detected elements of a given type plays no role in the identification procedure — only the specific combination of different types. All possible combinations of detected-element types must correspond to one and only one entry in the identity matrix. The entry in turn specifies a stylized-unit type. For nondifferentiating sensors, the expected unit identities may be less descriptive (e.g., unidentified unit with wheeled vehicles). These identifications would be one of the last 10 general unit types. Although the program is not presently implemented, it is a minor task to allow certain sensors to identify on a less rigorous basis (e.g., Radio DF can be made to identify units on the basis of detecting radios). Once a unit is given a stylized identity, unit size and target-detection probability are computed for the sensor. The expected numbers of detected elements then become important. The expected size of the target unit is computed by comparing the expected number of detected elements of each type with the total number of that type in each of the three sizes of the stylized unit. The binomial probability of detecting the expected value of elements out of the number available in each size of the identified, stylized unit is summed over element types. This will produce three probabilities — one for each unit size. The size having the highest probability is designated as the expected size for the target. Again, a simple modification would allow certain sensors to have more accurate sizing capability for special target elements.

The unit's detection probability is computed on the basis of a minimum-number parameter for each element type (input by the user). This minimum number represents a lower limit of element detection in that any number less than this would not be considered to be a target of any importance. The binomial probability of detecting a number of elements less than this minimum number is computed for each target-element type. (This is merely the sum of all binomial probabilities from zero to the minimum number, given the number available in the target and the single-element-detection probability.) The product of these binomial sums for all present element

types is determined to be the probability that the unit is *not* detected. Hence, the unit's detection probability is simply one minus this product.

This entire identification, sizing, and probability calculation is perhaps best illustrated by an example.

Assume that one is using the element and unit types shown in Appendix G and consider the following situation: One of the differentiating (capable of precisely identifying target elements) sensors is interacting with a HAWK battery in the scenario. Suppose that under the conditions of the interaction (range and environment), the sensor has nonzero, element-detection probabilities for support vehicles, missiles, and radars (element types 4, 6, and 11). Call these probabilities P_4 , P_6 , and P_{11} . Hence, nonzero expected values of detection would be computed for these element types — say N_4 , N_6 , and N_{11} . This would correspond to element-type combination (4, 6, and 11) which in turn would represent an element in the identity matrix. This entry would specify stylized-unit type 4 (air defense). The target would then be so identified. Note that this target could have been misidentified if one of the crucial elements would have had zero or low probability of detection. For example, if the radars would not be found, only the combination (4, 6) for support vehicles and missiles would be considered. This combination (4, 6) could correspond to a different entry in the identity matrix, probably unit type 8, missile artillery.

Under the conditions specified, however, the unit would be correctly identified as air defense; hence, stylized-unit type 4 would be consulted.

There must be three organizational sizes:

- Firing section containing 10 personnel, 1 support vehicle, 1 missile, and 1 radar.
- Battery containing 98 personnel, 35 support vehicles, 3 missiles, and 4 radars.
- Battalion containing 496 personnel, 174 support vehicles, 12 missiles, and 16 radars.

Recall now that the given sensor has the following single-element probabilities and expected number of detected elements:

	<u>Single-Element Probabilities</u>	<u>Expected Detections</u>
Personnel	0	0
Support Vehicles	P_4	N_4
Missiles	P_6	N_6
Radars	P_{11}	N_{11}

The probability that the unit would be perceived as a section is

$$P_{s1} = \text{Binom}(N_4, P_4, 1) + \text{Binom}(N_6, P_6, 1) + \text{Binom}(N_{11}, P_{11}, 1)$$

where $\text{Binom}(x, y, z)$ is the probability that one will detect x objects out of z available when the single-element detection probability is y .

Similar probabilities are computed for battery and battalion (P_{s2} and P_{s3}). The size unit which has the greatest probability is designated as the expected size of the unit.

The probability that the unit is detected is computed by considering the input minimum numbers of detected elements for target consideration. Assume the following numbers have been input for minimums:

20 personnel
10 support vehicles
1 missile
1 radar

The probability that the detected elements would not be considered a target is the probability that, under the conditions, less than the above specified minimum of each type will be detected. The probability that less than the mini-

um number of support vehicles (10) will be detected is $P_{L4} = \sum_{i=0}^9 \text{Binom}$

(i, P_4, N_4) where N_4 is the number of support vehicles in the real target (in this case 35). Similar probabilities are computed for personnel ($P_{L1} = 1$), missiles (P_{L6}), and radars (P_{L11}). The probability that the target will not be detected is $P_{(\text{not detect})} = P_{L1} \cdot P_{L4} \cdot P_{L6} \cdot P_{L11}$. Hence, $P_d = 1 - P_{(\text{not detect})}$.

The target-definition section also computes the estimated, target-location error by consideration of the sensor CEP for each of the element types and a centroid error. Centroid error is error due to the fact that located target elements need not have the same centroid as the true-target centroid.

Although the target-definition methodology has some weak points, especially in the target-sizing aspects, the fact remains that this model has the only formal methodology among the four models under consideration. With a few modifications, an experienced (and clever) user can apply this methodology very efficiently and successfully if sufficient one-on-one data exists.

(6) Total-Target Detection Probability. The CAMWTH model does not compute the total-detection probability for a target. However, since the fire planning and damage calculation routines require a detected-target list, one is generated by the model. Each target which interacts with any sensor in the input sensor array will be placed on the detected-target list. If more than one sensor in the array detect a given target, the selection of which sensor report to use is based on one of three user-specified criteria:

- Greatest, estimated-target size.
- Smallest, target-location error.
- Highest targeting priority.

For each sensor which interacts with a given target, the target-definition routine will compute the target-detection probability, estimated-target identity and size, and location error for that particular sensor. This type of information will be used for damage calculations and fire planning and will critically affect the results. Reports produced by all sensors will be tested against one of the above criteria, and the sensor report which is added to the target list is the one which best satisfies the criterion. All reports from other sensors are completely ignored for that target. The fire plan and the damage calculations are based solely on the one selected report.

In a sense, since the target array is played against a sensor array, CAMWTH's target-acquisition module can be considered a many-on-many model. However, the method by which multiple-sensor reports are treated can in no way be considered a valid treatment of total-target-acquisition probability.

This is unfortunate because the information is there for the asking. It would be a trivial task to compute the target's total probability of being detected and the spectrum of probabilities for the various classifications of the target given detection. This could be done by computing a set of weighted values for the targets estimated identity, size, and location. The inclusion of time may be a more difficult task but it is certainly not insurmountable. The use of these figures for later fire planning and damage calculations would add to the validity of the results although this may be more difficult to achieve. However, whether or not such information could be used by CAMWTH's later modules, these modifications would make CAMWTH an attractive choice for the TNF target-acquisition model. One can easily imagine a scheme for producing a more generalized and valid (although it may not be pure expected value) acquired-target list.

(7) Other CAMWTH Features. The prime purpose of the CAMWTH model is to evaluate the casualties inflicted on friendly (BLUE) forces by hostile (RED), indirect-fire weapons systems. Hence, in addition to target acquisition, the

model must evaluate the effect on friendly targets when acquisition has occurred. This is done by CAMWTH in the three modules (MOD2, MOD3, and MOD4) following the target-acquisition module (MOD1).

In brief, MOD2 (consisting of two possible approaches: MOD2C for conventional weapons, and MOD2N for nuclear weapons) does target analysis on the detected-target list. This module computes the expected damages which would occur if enemy fire units engage the perceived target. Each perceived target is played against each enemy delivery system, and the expected damage is computed and stored for use in later fire planning. The detected-target list includes only the data which the enemy, target-acquisition system would have perceived (not necessarily the correct information). The methodology used in the damage calculation is purely expected value and is based on approved Army methodologies. For nuclear weapons effectiveness (MOD2N), the methodology implemented is that in FM-101-31-2, Staff Officers Field Manual; Nuclear Weapons Employment Effects Data (SECRET). The conventional munitions effects data (MOD2C) was derived from the Joint Munitions Effectiveness Manuals. Both MOD2C and MOD2N are modifications of a program called FEBSAB developed by the US Army Combat Developments Command Institute of Nuclear Studies.

The fire-planning module (MOD3) specifies which fire units and delivery systems will be assigned to engage which targets. The fire planning uses the results of the target-analysis module and assigns targets to fire units on the basis of one of three user-specified criteria:

- (a) Low-cost, high-collateral damage, most-direct control.
- (b) Low-collateral damage, low-cost, most-direct control.
- (c) Most-direct control.

Fire units satisfying the criterion and defeating the target are assigned on a first-come, first-served basis from the detected-target list until all fire units are completely occupied for the duration of the engagement or all defeatable targets are engaged. The module does not consider the actual detection probabilities or target priorities.

MOD4 is the fire-plan execution module. It is identical to MOD2 except that only the fire units specified by the fire plan (output of MOD3) will engage the real targets. Note that the fire plan is made using the estimated target array and is executed using aimpoints and munitions specified by the fire plan. However, the expected damages are computed using the real target's location and composition.

After all four modules are executed, a summary of all casualties (both men and equipment) is output by the executive.

(8) **CAMWTH Summary.** On the whole, CAMWTH has the capability to perform all of the required functions of a target-acquisition model. The model considers adequate scenario detail for both the friendly (BLUE) target array and the hostile (RED) sensor array. There is also adequate flexibility for user manipulation of the scenario. There is a consideration of one-on-one interactions (i.e., individual sensors versus target elements) via the probability tables. It is the only model for which there are formalized, target-definition calculations. It has the capability of computing total-target-detection probability and generating detected-target lists although modifications are necessary in this area. In addition, it does consider weapons delivery and damage calculations if desired.

There are some modifications which should be made to make CAMWTH completely responsive to the TNF/S study. There are also some desirable modifications which would enable one to use the model to its full capability. These are outlined below. The first set listed are those modifications which should be considered first because they are either more important or easier to implement. The second set of modifications should be considered, but they are probably more difficult to implement and not as essential.

(a) **Recommended Modifications:**

1 Modify the sizing and target-detection-probability calculation to include multisensor interactions. That is, incorporate a more valid many-on-many methodology. These calculations should produce a weighted spectrum for the identity, size, and location error for a given target for the combined sensor array. For each target, the combined sensor report should consist of the following:

- Probability of Detection (combined for all sensors).
- Possible identities with associated probabilities.
- Possible sizes with associated probabilities.
- Location errors with associated probabilities.

It may be easiest to do this separately for all sensors of a given type and then to combine sensors of different types for a detected-target list either external to the model or with a different model.

2 Although the handling of time in scenario considerations is adequate, it is not so (hardly even considered) in the individual target-sensor interaction. Some considerations must be made to include the effects of time on

element detection (i.e., consider the total time of target-sensor overlap in the detection calculations).

3 Give certain sensor-element pairs special considerations in the identification and sizing calculation. In a sense, one should consider certain element types (e.g., radios) as having distinctive signatures which enable certain sensors (RDF) to classify their parent target unit more efficiently. This would remove these special sensor-element pairs from the normal target definition tree. There are other similar pairs such as radars — ELINT systems and cannons — sound-ranging systems. These special pairs can have different capabilities of classification.

4 Include target location with respect to the FEBA in the identification decision.

(b) Desirable Modifications:

1 Increase the number of allowable-element types. This would help in better tailoring unit identification since there could be more key-element types which identify units. This, however, could present a few problems in the identification routines, and it also complicates the problem of excessive data requirements. The use of special-object types (e.g., radios) for certain sensor types may alleviate these problems somewhat.

2 Change method for considering target-sensor pair interactions. In general, cycle each target on the individual sensor and store sensor reports on mass storage for later combination into a single intelligence report on each target. This should give the user a handle on sensor saturation.

3 Devise method to allow different levels of discrimination for various element types (e.g., for element-type missile, allow finer discrimination into specific type based on target-unit type such as HAWK missile, NIKE missile, LANCE missile). Allowing a larger number of element types could probably accomplish the same goal. However, the use of two discrimination levels for elements is likely to make the target-identification process appear more realistic.

4 The fire-planning and damage-calculation modules should be changed to incorporate the sophisticated identity and sizing predictions which would be available if modification (1) of the first set of modifications is implemented. This would not affect the target-acquisition module.

If most of the modifications in the first suggested set are implemented, CAMWTH would be an excellent choice for the TNF. The model could

be used independently or could be supplemented by superior one-on-one models for data or a better many-on-many model for a detected-target list.

c. The Science Applications, Inc. (SAI), Combat Survivability Model.

(1) Overview of the Model. The SAI Combat Survivability Model (COMBSYS) is primarily a nuclear weapon's effect model. The model computes the damage incurred by an input, real-target array when engaged by a specific weapon's allocation. Both the target-acquisition and weapon's allocation methodologies were originally performed external to the main model. The target-acquisition methodology consisted of an analysis of various operational, sensor, and target parameters against a given target list. The results of the analysis are the expected number of targets (by type) detected in a specific period of time. The methodological approach is primarily one which was outlined in a paper "Demonstration of a Method for Determining Target Acquisition Capability" by W. R. Schilling of SAI in November of 1973.

The weapon's allocation model uses the expected number of targets of each type and the relative value of targeting priority of each type to determine an allocation of weapons-to-target type which maximized the total, expected value destroyed.

After the allocation model has been exercised, a MONTE-CARLO process is used to generate a specific, detected-target list and specific allocation of weapons (i.e., specific aimpoints and burst times). These are then played in the COMBSYS model to compute the damage effects of an attack.

The target-acquisition methodology has since been modified and computerized (for Harry Diamond Laboratory). This new, stand-alone model is called "The SAI Target Acquisition Model," "The SAI Mobile Target Acquisition Model," or, simply, "The SAI Model." Only the methodology, inputs, and outputs of this model will be considered further.

The SAI model is a good example of a pure, many-on-many model. The model computes total-target-acquisition (defined as detection and classification) probability using input values for single-sensor-type, detecting targets. This acquisition probability is computed for each individual target in the input array.

By using these total-acquisition probabilities, a MONTE-CARLO procedure produces a list of detected targets. The detected-target list is a representative sample of the real-target array weighted by the total-acquisition probabilities.

The model methodology is quite straightforward. Primarily, the model computes the time-adjusted, aggregate probability of detecting a given target using many, individual, single-sensor probabilities. (All are assumed to be operating independently of each other.)

With one exception (the COMINT System), the model has no explicit one-on-one models or one-on-many models. The important one-on-many aspects are input to the model. Although it cannot really be considered a one-on-one model, there is a separate, more-detailed model for the COMINT System. The output of the COMINT model is the probability that a COMINT target would be detected. This probability is a direct input to the main SAI model. The COMINT model will be briefly described in the total-target-detection section.

Although called the "Mobile Target" model, the SAI model computes acquisition probabilities for pseudo-stationary targets (those that move only occasionally). Continuously moving targets are handled differently by the model. Essentially, total-acquisition probabilities must be computed or estimated external to the model for continuously moving targets. These probabilities are input to the model, and these moving targets are considered for inclusion in the final, detected-target list.

The inputs to the main model are summarized in Appendix I and the outputs, in Appendix J. Details concerning the data will be further described in later sections.

(2) Scenario Methodology. Similar to all the other models, the friendly (BLUE) scenario is specified by a specific target array on a map grid. Unlike the other three models under consideration, SAI model targets are not collections of collocated elements. They are simply specific unit types (e.g., infantry platoons, tank platoons, command post).

Targets are subdivided into an arbitrary number of classes. All targets in the same class have common target values (priority as a target), acquisition properties, and target permanence.

Generally, all targets of the same unit type are in the same class, but this is not a requirement. It is possible to consider targets of similar unit types as belonging to different target classes if their posture, target value, acquisition characteristics, or location differed. For example, two target tank platoons, one engaged near the FEBA and the other held in reserve far behind the FEBA, can be in two different target classes.

After targets are sorted according to class, the classes are themselves combined into a lesser number of acquisition types. With the exception of SIGINT considerations, all target classes of a given acquisition type have identical vulnerability to detection.

The SIGINT vulnerability is considered separately for each class. It should be noted that the vulnerability to detection of moving targets is a separate input parameter and is independent of the kind of target class or acquisition type the target belongs to.

Stationary targets are defined by their grid location in the map scenario and their target class. Moving targets are defined by their class, starting point, times of movement, average velocity and direction of movement, uncertainty of velocity and direction, and the total-acquisition probability for the specific targets.

The size of the target array can be quite large. This is not a large user burden because the amount of data required per target is small. The data requirement can become very large with increasing numbers of target classes and sensor types because probability data must be supplied for all applicable acquisition-type/sensor-type pairs. However, it is believed that extremely large scenarios can be handled without great difficulty.

The model handles time adequately if time slices are short enough so that the majority of targets can be considered stationary. However, the scenario writer must be cautious concerning temporal matters. Time can have a critical influence on the acquisition probabilities, and there are several inputs (e.g., target permanence and activity factors) which may have to be adjusted depending on the length of the scenario.

(3) Surveillance Methodology. The sensor-system array in the SAI model is specified in terms of available assets of each sensor type. There is no specific deployment of sensors on the map grid. At present, there are eight types of sensors modeled:

- Counter Battery/Counter Mortar Radars (fixed system).
- Sound Ranging (fixed system).
- Forward Observers (fixed system).
- Penetrating Aircraft with photographer, visual or IR systems (Penetrating Sweep-Rate System).
- Airborne Surveillance from Hostile Side of FEBA (Sweep-Rate Systems Flying Along FEBA).
- Patrols or Agent.

- ELINT Systems.
- COMINT Systems.

The probability-of-acquisition calculation varies from sensor type to sensor type. There are six different methodologies used — a different methodology for each of sensor types 4 through 8 and a methodology for fixed systems (sensor types 1 through 3). These calculations will be considered in more detail in later sections.

The number of sensor types can be easily expanded. For example, different, individual sensor types can be assigned for different kinds of aircraft reconnaissance (e.g., airborne photo reconnaissance can be different than airborne IR reconnaissance); similarly, for forward observers or patrols. Completely different kinds of surveillance such as satellite reconnaissance could also be included. These changes would require code modification, but the modifications should not be difficult.

The model computes the acquisition probability for each target in the array by each sensor system individually. This calculation includes consideration of the total number of sensors of that given type and the time (number of looks) that the sensor system would have to detect the target. The total-acquisition probability of the given target is then simply the combined probability of all sensor systems assuming that each of the individual sensor system's probabilities is independent. In other words, there are no communications between sensor systems (or even between sensors of the same sensor type). There is also no explicit consideration of possible sensor saturation. These effects must be considered externally by modification of input-probability curves. The resultant acquisition probability is the probability that the given target would be detected by at least one sensor during the modeled time period of the exercise. There is no consideration of possible loss of contact (i.e., a sensor detects a target, but target contact is lost due to target movement or sensor destruction).

The detected-target list is determined by first ordering the targets in the array by their acquisition probabilities and then dividing the array into groups with similar acquisition probability. The expected number of targets from each group is then computed and rounded to the nearest integer. Targets are then chosen at random from each group so that the number of expected targets from each group is satisfied.

Detection times for each detected target are then assigned from a random sampling of a rectangular time distribution over the total time period of the exercise.

(4) **One-on-One Methodology.** There is no one-on-one methodology in the SAI model.

(5) **Target-Definition Methodology.**

(a) **General Capabilities and Input Requirements.** The SAI model has no explicit, target-definition routines. This necessary target-definition function must be done externally, and the data must be input to the model. The model requires input data which would normally be the result of target-definition methodology. This data is in the form of range-dependent curves for single-glimpse probability of detection (P_d), probability of classification given detection ($P_{c/d}$), and time to detect and classify ($T_{d\&c}$). The actual form of these curves varies for different sensor types.

For sensor types 1 through 5, there are three curves (one for each of three possible concealment modes) for each of the three quantities (P_d , $P_{c/d}$, and $T_{d\&c}$). This data must be supplied for each applicable acquisition type. This is a total of nine curves per applicable acquisition-type/sensor-type pair. (Note: this data is not necessary for some pairs, specifically, those pairs where targets in the acquisition type do not emit a signature relevant to the particular sensor type.)

For sensor type 6 (Patrols), these curves are reduced to a single number (a single-glimpse probability of acquisition). That is, $P_d(R) = \text{constant}$, $P_{c/d} = 1$, and $T_{d\&c} = 0$. The single-glimpse probability of acquisition must be supplied for each acquisition type and each cover and concealment mode.

For sensor types 7 and 8 (SIGINT sensors), target-acquisition types and cover-and-concealment modes are not considered. Each target is individually specified as being an ELINT target, a COMINT target, or neither. COMINT targets all have the same fixed, COMINT-acquisition probability determined externally to the model. For ELINT sensors, the same three nonrange-dependent numbers (P_d , $P_{c/d}$, and $T_{d\&c}$) are used for all ELINT targets.

Target-location error is input for each acquisition type in each range band. The same location error is used for all non-SIGINT sensors. Location errors for SIGINT sensors are separately input for COMINT and ELINT; both are range-band dependent. The target's location error is a weighted average (based on acquisition probability) of the non-SIGINT and SIGINT errors.

In addition to the large quantity of data necessary, the probability curves and location errors are not well known. This is a serious deficiency of the SAI model. With the exception of COMINT, which has a detailed, external

model, it is difficult to trace model results back to specific field test data or detailed technical sensor models because there is no well-defined, target-definition methodology. This deficiency seriously limits the use of this model. The model should not be considered a good candidate for TNF/S by itself unless extreme simplicity is desired.

(b) **COMINT Submodel.** The COMINT submodel computes the probability that the net control station of "important" communication nets will be detected and located during the time of the exercise. The model is a Monte-Carlo simulation of the temporal processes which must occur for COMINT acquisition. It is not a technical model of the propagation of signals, probability of intercept, or location errors of COMINT-type intercept hardware.

The model considers:

- Hostile intercept assets in terms of number of intercept monitor stations and DF stations.
- Probability of intercept of friendly radio transmissions (based on number of friendly communications nets and radio traffic per net).
- Time delays which occur in the COMINT process.
- The interrelationship between these time delays and the acquisition probability of important net control stations.

COMINT system time delays include:

- Master Intercept Monitor Station assignment of frequencies.
- Intercept Monitor Station recording and transmission delays.
- Intercept Analysis.
- Recon BN HQ delays.
- DF station and mission assignment.
- Division HQ Target-assignment delays.

All temporal data are input as average values with associated uncertainties.

The Monte-Carlo approach simulates the COMINT process by generating a time sequence of intercepted transmissions from each net and then processing the intercepted information through the time delays to determine how many net control stations are fully acquired during the time period of the simulation. The number of stations acquired divided by the number available is the acquisition probability for these COMINT targets. This acquisition probability is averaged over many Monte-Carlo cycles. The average acquisition probability is then used in the overall SAI target-acquisition model for the COMINT probability of acquisition for the

specified COMINT targets (those targets which contain at least one communication net control station).

(6) Total Target Detection. Computing the total-target-acquisition probabilities and subsequently using these probabilities to generate a representative, detected-target list are the sole considerations of this model. The generation of the detected-target list was described earlier. The calculation of the total-acquisition probability for each individual target is quite straightforward.

For non-SIGINT sensors, the probability that one of the individual sensors of a given type will acquire the target in a single glimpse is computed for each cover-and-concealment mode. The single-glimpse probability is then adjusted for multiple glimpses by all sensors of that given type for the target in the given, cover-and-concealment mode. The total probability for all non-SIGINT sensors is then the combined probability of all the sensor types weighted by the probability of the target being in each cover-and-concealment mode.

SIGINT sensors are handled separately. First, a target is either an ELINT target or a COMINT target or neither. No consideration is made of the cover-and-concealment mode. All ELINT or COMINT targets are treated with range-independent probabilities. With the above exceptions, ELINT sensors are treated very similarly to fixed, non-SIGINT targets for the SIGINT component of its acquisition probability. However, COMINT targets are all treated with a single, fixed-input, SIGINT-acquisition probability which is computed externally with the COMINT submodel.

The total-acquisition-probability calculations are not complex and are all illustrated as follows (note that all defined quantities must be specified by input data):

General Information

- R = Range.
 - T = Total exercise time.
 - WF = Target array front dimension.
 - i = Sensor type ($i = 1, 8$).
 - j = Target-acquisition type ($j = 1, NA$).
 - k = Cover-and-concealment modes ($k = 1, 3$).
- Fixed Sensors ($i = 1, 3$).

Have input curves (range dependent) for:

- $P_c(i, j, k, R)$ = Single-glimpse probability.
 $P_{c/d}(i, j, k, R)$ = Probability of classification given detection.
 $T_{d\&c}(i, j, k, R)$ = Time to detect and classify (and report to FDC).
 $P_{los}(i, R)$ = Probability of line of sight.

Also, have values for:

- $N(i)$ = Number of sensors of type i.
 $SW(i)$ = Sensor front (dimension of sensor front for individual sensor of type i, parallel to FEBA).
 $Av(i)$ = Fraction of the time an individual sensor of type i will be available (or active).
 $Per(j)$ = Target permanence for target-acquisition type j (expected time of deployment).
 $Act(i, j)$ = Fraction of the total exercise time that targets of acquisition type j will emit signatures which can be detected by sensor types i.
 $Vis(i, j)$ = Attenuation factor for sensors of type i looking at acquisition type j.
 $C(j, k)$ = Probability that targets of acquisition type j will be in cover-and-concealment mode k.

Now, for each target, the acquisition type j is fixed (since it must be in one of the target classes with a fixed-acquisition type). The range from the FEBA is also fixed by its deployment. Hence, the range-dependent quantities, P_d , $P_{c/d}$, $T_{d\&c}$, and P_{los} , are computable. Thus, the probability (P_{sik}) that a single sensor of type i will acquire the target on any given glimpse when the target is in one of the cover-and-concealment modes (k) is, therefore:

$$P_{sik} = P_d \cdot P_{c/d} \cdot P_{los} \cdot Vis(i, j) \cdot F$$

where F = probability that the target will be available for sufficient time to be classified and reported after detection.

$$F = \begin{cases} \frac{\bar{T} - T_{c\&d}}{T} & \text{if } T \geq T_{c/d} \\ 0 & \text{otherwise.} \end{cases}$$

$$\bar{T} = [\text{Min } T, Per(j)].$$

Now, the expected number of glimpses for the fixed sensor i is

$$Ng(i) = \frac{N(i) SW(i)}{WF} Act(i, j) Av(i).$$

Hence, the probability that the sensor i will acquire the target in cover-and-concealment mode k during the time period of the exercise is

$$P_{ik} = 1 - (1 - P_{sik})^{Ng(i)}.$$

Sweep-rate sensors:

The same as above, except that for penetrating sensors ($i = 4$),

$$Ng(i) = \frac{N(i) \cdot V(i) \cdot \bar{T} \cdot PS(i) \text{Act}(i, j) \cdot Av(i)}{A(i)}$$

where $V(i)$ = Sensor velocity
 $A(i)$ = Search area
 $PS(i)$ = Mission survivability;

and for sensors flying parallel to FEBA ($i = 5$):

$$Ng(i) = \frac{N(i) V(i) \bar{T} PS(i) \text{Act}(i, j) Av(i)}{WF};$$

Patrols ($i = 6$):

N_p = Number of patrols
 A_p = Area searched per unit time by one patrol
 A_T = Total area to be searched by patrols
 P_{sik} = Single-glimpse probability of acquisition for target in cover-and-concealment mode k

$$Ngp = \frac{N_p \cdot A_p \cdot T}{A_T} = \text{Number of Glimpses.}$$

$$P_{ik} = 1 - (1 - P_{sik})^{Ngp}.$$

Hence, the probability for all non-SIGINT sensors is, therefore:

$$P_{NSIG} = 1 - \sum_{K=1}^3 C_{(j, k)} \left\{ \prod_{i=1}^6 (1 - P_{ik}) \right\}$$

and, since j is completely defined for each target, P_{NSIG} is also defined.

SIGINT sensors ($i = 7, 8$):

ELINT ($i = 7$):

All ELINT sensors are assumed to be ground-based DF (negligible modification to allow other forms). Calculation is the same as for other fixed sensors except that there is no dependence on range or cover-and-concealment mode.

COMINT ($i = 8$).

The probability of acquisition for *all* COMINT targets is input as a single number. The number is derived from independent COMINT submodel.

The SIGINT acquisition probability is then:

$$P_{\text{SIG}} = 1 - (1 - P_{\text{ELINT}})(1 - P_{\text{COMINT}}).$$

The total-acquisition probability for the target is then:

$$P_{\text{ACQ}} = 1 - (1 - P_{\text{SIG}})(1 - P_{\text{NSIG}}).$$

This acquisition probability is computed for each target in the input array and then used in the generation of the detected-target list.

The SAI model also determines an estimated, target-location error for each target. The error is simply the weighted average of non-SIGINT (input by range band and acquisition type) and SIGINT (input by range band) errors. The weighting factor is the probability of acquisition.

Of the four models under consideration, the particular procedure of the SAI model for total-target-detection calculation is unique and is the only treatment of the total-target-acquisition function with any validity at all.

(7) Other SAI Features. The SAI Target Acquisition Model is a computerized version of the target-acquisition methodology used for the SAI Combat Survivability Model (COMBSYS). Although the two models are not presently combined into a single model, it is not difficult to form a combined, more-elaborate model involving target acquisition, weapon's allocation, and target-damage calculation for the analysis of theater nuclear or nonnuclear force survivability. SAI already possesses modules which use the output of the present, target-acquisition model (a detected-target list) as input to weapon's allocation.

(8) **SAI Summary.** The SAI model is the only example of a many-on-many model. However, that is all it is. It has no one-on-one methodology and no explicit one-on-many methodology. The required inputs include data which might be considered the results of a one-on-many model. That is, the model requires the probability that a given, individual sensor will detect a target (not just target elements) and then identify the target as a specific unit type and size. The data requirements are sizable considering how few calculations are actually performed.

The SAI model is the only model of the four under consideration which specifies the sensor scenario in terms of available sensor assets as opposed to specific deployments of the sensor array. While this approach has the advantage of removing specific sensor deployment as a variable, it also means a loss of many important, temporal considerations. This coupled with the fact that the user must input acquisition probabilities for moving targets, makes it difficult to model dynamic situations. Thus, time slices must be relatively short. Another problem in the handling of time is the fact that BLUE targets are defined in terms of their target permanence at a given location as opposed to specific time periods of deployment. Since target permanence is associated with the target's acquisition type, this removes some of the flexibility in scenario writing. Since detection times are randomly assigned to targets in the detected-target list, there could be significant problems if the detection time of specific targets were a critical factor. The model does consider time adequately within the context of the model's original intended use; however, for long, dynamic scenarios, the model user must be very careful with the temporal variables. Otherwise, very erroneous acquisition probabilities may be computed.

The model's simplicity and quick execution times are very attractive features. In addition, the fact that this model computes the overall probability of acquisition for different types and range bands makes it unique when compared with the other models. However, the procedure used is easily applied to the other models. It should be possible to compute a time-sequenced history of total-acquisition probability for each target using a modified version of this approach.

Because of the requirement of rather uncertain data, the problems in handling matters of time, and the lack of connection with field-measured or technically computed probabilities, this model should not be considered a good candidate for the TNF unless extreme simplicity is desired. The total-target-acquisition methodology can easily be applied to other models if one is willing to accept the lack of sensor synergism between different individual sensors.

d. The STANO-SAM IV Model.

(1) Overview of the Model. The Surveillance, Target Acquisition and

Night Observation-System Assessment (STANO-SAM IV) Model is the most recent version of a large surveillance-simulation model developed by the CALSPAN Corporation. The original version (STANO-SAM I) was completed in 1971 for the U.S. Army Combat Developments Command. Although it included three generic-attended sensor types (Thermal Imaging, Near IR and Visual Imaging, and Radar), the unattended sensors were the driving force in its development. In 1973, USAECOM sponsored additional development to incorporate a classifying logic into some of the unattended sensors. This became STANO-SAM II. Other significant changes were completed in 1976 resulting in the STANO-SAM III (also for USAECOM). Version IV is little different from its predecessor; the only difference is the inclusion of a CB/CM radar sensor. Hence, STANO-SAM III is essentially the present version of the model.

The model is an extremely complex, enormously detailed stochastic simulation of surveillance and target acquisition on a brigade-size area (30 km by 30 km). The simulation includes both attended and unattended, ground-based sensors and airborne sensors. The "targets" which interact with the sensors are groups of men, boats, aircraft, tanks, and trucks. Also included are false targets such as nonmilitary vehicles or personnel, animals, and natural or artificial disturbances (artillery fire or thunder) which can cause false sensor alarms.

The overall model is a collection of several submodels each of which is run with separate job steps and requires separate input. (Often, the input includes the output of a previous segment of the simulation.) Source programs for all the submodels consist of over 300 FORTRAN subroutines coded with approximately 50,000 statements.

There are two major subprograms which make up the core of the model. One is called PRERUN which is responsible for the following:

- (a) Card input for the scenario (targets and sensors).
- (b) Pre-processed data for environmental data, LOS data, etc.
- (c) Plotting of input data for a user check.
- (d) Modification and reformatting of sensor-system parameters for later use.
- (e) Time sequencing of various events which occur during the simulation:
 - Sensor-Target Interactions.
 - False Alarms.
 - Sensor, data link, monitor changes (includes parametric changes, up/down times, emplacement, and removal).
 - Battlefield Illumination.

The output of PRERUN is then the input (via disk) into the Main Simulation Model (MSM). The MSM is the portion of the model which calculates the result of each target-sensor interaction specified by PRERUN. The output of the MSM is the time history of various, sensor-detection results. This output is in the form of printer output and a binary file (disk or tape) called MSMOUT.

In short, PRERUN is responsible for all scenario activity while the MSM performs the detailed sensor physics calculations to compute detected elements.

There are several subprograms necessary for processing input before using PRERUN or for illustrating the raw input data. These subprograms are described as follows:

- Atmospheric Model which generates detailed atmospheric data over the entire area of operations (battlefield). The data is generated as a function of time and is used by PRERUN, MSM, and other two-input processing programs. The program also does some bookkeeping manipulations for vegetation ground cover and micro-terrain tables. These tables are also used by later programs.
- Terrain Model which generates a digital terrain tape to be used by PRERUN for LOS determinations and the other two-input processing routines.
- Radar Contour Plot Program which uses the digital terrain data and atmospheric data to plot the coverage area for selected radars in the scenario. (NOTE: This program is not needed for PRERUN execution.)
- RF Data Link Analysis Program calculates transmission for given transmitter and receiver locations. The model considers both foliage and terrain effects and atmospheric effects (mainly, unattended sensor arrays with monitors). (NOTE: This program is not needed for PRERUN.)

After MSM has executed, there are three subprograms which perform additional analysis on the MSM binary output:

- Unattended Sensor Analysis Model (ANALYZE) is a simulation of the inferences and errors which might be made about target presence, speeds, direction, and number and type of elements. The model computes the perceived time of arrival of targets at firetrap points.
- Tactical Communications Model calculates the time delays of messages originating at sensor stations and sent to various Headquarters.

- Attended Sensor Model simulates the reports which would hypothetically be made by attended sensor operators. The model attempts to predict which of several stylized messages the operators may send. The reports are limited to types and numbers of target elements. No attempt is made to identify type or size of target unit.

- Utility Programs to *selectively* extract data from binary file MSMOUT in order to obtain time histories and statistical data. NOTE: The present binary output from MSM may not be in the correct format for direct use by some of the post MSM subprograms. Hence, modification of these subprograms may be necessary.

The existence of the MSMOUT file also gives the user the capability of further extension of the overall model capability. Driven by MSMOUT (and other auxiliary data), other post-processing programs can be devised which use various methodologies to provide computer support for analysis of MSM results.

Figure 7 illustrates the job steps and data flow between the current subprograms.

Appendices K and L provide brief overviews of the input data and outputs, respectively, of all subprograms except PRERUN and the MSM. The tabular information is segregated by subprogram.

For the purpose of the model comparison, main emphasis will be placed on the investigation of how the various portions of STANO-SAM can handle the (BLUE) target scenario, the (RED) sensor scenario, and the one-on-one models. These are the activities simulated within the PRERUN and MSM submodels. These two subprograms make up the bulk of the model. Hence, the remainder of this section will deal with the functions of PRERUN and the MSM.

PRERUN is the first of the two primary STANO-SAM subprograms. Its purpose is to simulate all activities associated with the scenario and sensor-system operation excluding the detailed sensor physics simulations. PRERUN is divided into the following 13 job steps (labeled as Job Step 0 through Job Step 12):

Job Step 0 – Accepts input data from all sources (card input and all data generated by the two data processing subprograms, TERRAIN and ATMOSPHERE). Converts data for later use by other PRERUN job steps. Checks data for internal consistency and plots sensor, target, monitor, and firetrap positions (if user specified).

Job Step 1 – This step computes the up/down times of monitors, data links, and firetraps. It also computes emplacement or removal times for

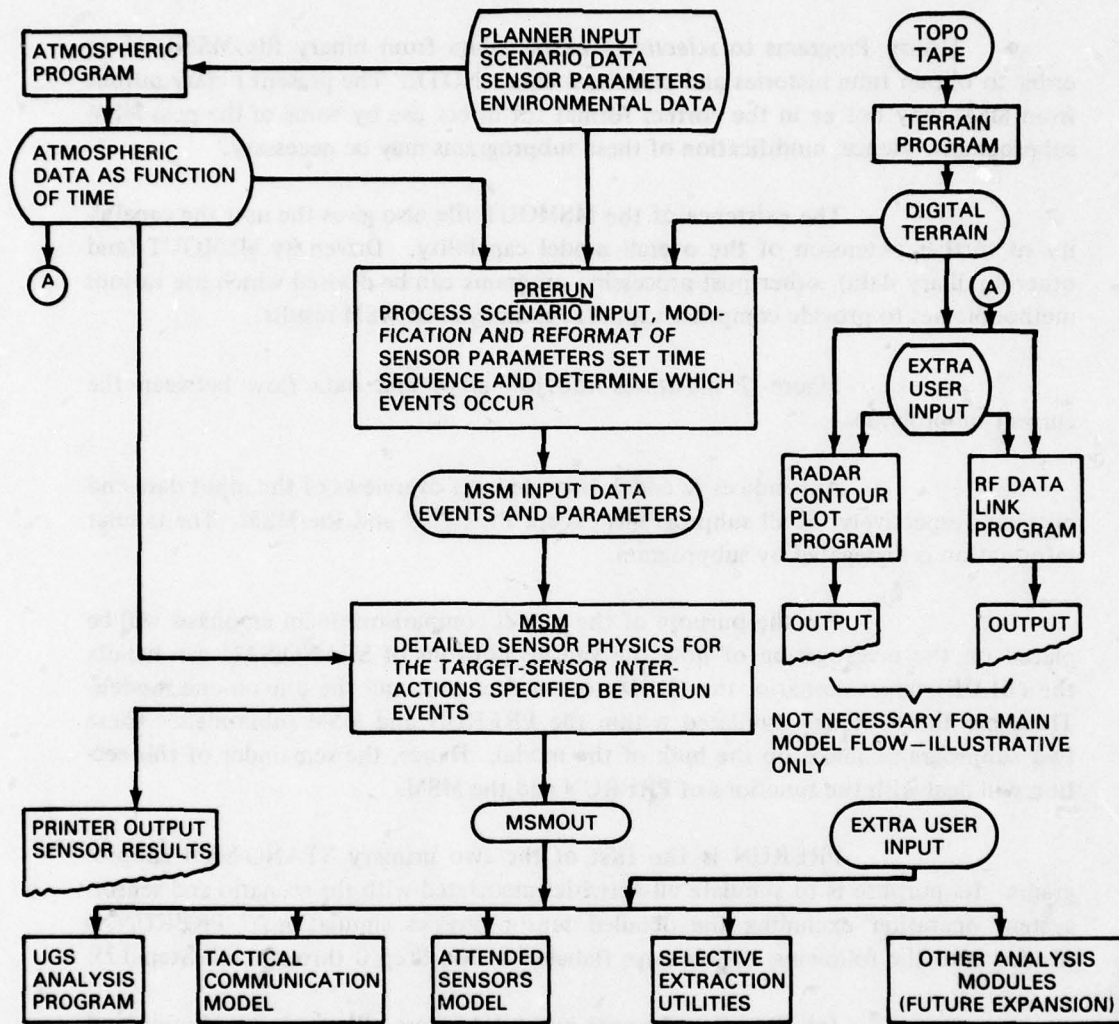


Figure 7. Job steps and data flow of the STANO-SAM-IV model.

unattended ground sensors. This data is stored both for later use by PRERUN and for output as events to the MSM.

Job Step 2 – This step computes up/down times of all stationary sensors (attended and unattended).

Job Step 3 – This step computes the ground-truth position of all stationary sensors.

Job Step 4 – This step computes the paths of all moving sensors and determines their up/down times.

Job Step 5 – This step modifies the sensor-system parameters and plots them for user check. The system data is then output to disk for use by the MSM.

Job Step 6 – This step creates battle and cultural noise level for later false-alarm generation. Step also stores battle-illumination events.

Job Step 7 – Computes data for sensor parameter changes, changes in atmospheric parameters, and background noise levels. This section also computes times for sensor false alarms.

Job Step 8 – This step computes all sensor-target events (excluding line-of-sight considerations). (NOTE: Some sensor platforms are also possible targets.) Only the possibility of sensor-target interactions is actually considered based on sensor coverage/target overlap and time.

Job Step 9 – This step is concerned with line-of-sight considerations. LOS computations are deterministic for large terrain features (from the input terrain data of the terrain subprogram) or probabilistic for micro-features.

Job Step 10 – This step generates the sensor-target interaction events for the MSM (includes LOS).

Job Step 11 – This step merges all possible MSM events in time sequence.

Job Step 12 – This step is the final formatting and output of all events for MSM.

Note that all external input is read during Job Step 0. However, temporary mass-storage files are created internally and used to transmit information between job steps.

The output of PRERUN consists of two, mass-storage files for later use by the MSM. One contains system parameters for the MSM sensor models while the other is a list of events to be processed by the MSM.

Appendix M briefly describes the inputs and outputs generated by PRERUN.

MSM is primarily a collection of routines for performing physical calculations on the input PRERUN events. There are three executive levels in program execution:

- Main Executive Routines (Level 1).

Group of routines which reads parameter data (outputs of Atmospheric Subprogram and parameter files of PRERUN) and event data from PRERUN file and then allocates event processing to level 2 supervisory routine.

- Event Routines (Level 2).

Group of processing routines which performs the required bookkeeping necessary for processing of the event and outputting the result. One of the level-2 supervisors is the sensor-interrogation routine which then would call a level-3 supervisory program for specific sensor.

- Sensor Routines (Level 3).

The specific, technical-sensor submodels which perform the detailed sensor physics for sensor-target interactions. These routines use environmental and system data which has been stored (with required updates as a function of time) by upper-level bookkeeping routines. Target detection results are determined on the basis of random-number draws compared to a probability, or whether a threshold signal is reached, depending on sensor type.

The program requires inputs from PRERUN (system parameters and PRERUN events) and the Atmospheric subprogram. These data are all supplied from mass-storage files. Very little card input is required (only a few header cards).

The output of MSM consists of both the printer listing and the primary file MSMOUT. These data are illustrated in Appendix N.

(2) **Scenario Methodology.** In the discussion of the previous three models, the friendly target array and how it is simulated by the model has been defined to be the BLUE scenario methodology. However, in STANO-SAM documentation, the surveillance forces are specified as BLUE forces whereas the target arrays are specified as RED forces. For the purpose of this model comparison, the terminology used for the previous three models will be maintained in the discussion of STANO-SAM (i.e., BLUE implies friendly target arrays and RED implies enemy sensor system).

In STANO-SAM, targets can be friendly units (BLUE target arrays), enemy (RED) moving sensors, or enemy units moving in the surveillance area. In general, only BLUE targets need be modeled for TNF purposes.

Targets are defined to be one or more elements of some single-element type. The allowable element types are:

- Personnel.
- Small vehicles.
- Heavy trucks.
- Tanks.
- Trains.
- Helicopters.
- Light aircraft.
- Jet aircraft.
- Rafts.
- Outboard Motorboats.
- PT boats.

The number of elements in any target is arbitrary.

In addition to the target's composition, the target's force type, times of existence, location on the map grid, and organization must be specified. For moving targets, the location is not specified as position on the grid but as route, direction, and velocity. The route must be one of those specified in the path data. The number of possible different routes is limited only by the dimension allowed for the path data. The target's force type describes certain properties affecting its detectability (e.g., ferrous metal content, concealment, spacing between elements, formation). The target's organization is a representation of the unit's organizational structure. Target organization plays no part in the simulation activities.

The physical characteristics of target elements pertaining to their detectability are all specified in the model code. There are no input parameters describing any physical aspects of the target elements. If one desires to alter the

element's characteristics (and, therefore, its detectability), it is necessary to change portions of the STANO-SAM source code.

Once the planner has supplied the required data for the target scenario, the model handles all of the remaining bookkeeping. The majority of this work is done by the PRERUN program. PRERUN determines whether a given target comes under the scrutiny of any of the enemy sensors, checks the line of sight, and then generates a sensor-interrogate event if appropriate.

Time is explicitly handled by PRERUN bookkeeping in great detail. Target and sensor positions are either directly input or are deterministically computed by PRERUN on the basis of input speeds and direction. This is an advantage of STANO-SAM as compared with SCREEN and CAMWTH. It is not necessary to simulate the path of target movement by the creation of additional "model targets." However, in order to simulate a stationary target deployment followed by a movement and new stationary deployment, it would be necessary to specify three model targets, one for each of the stationary deployments and one for the period of movement.

Other temporal considerations are even more important. STANO-SAM has very detailed and explicit characterizations of changes in environmental features, sensor and target parameters, and battlefield conditions. These can significantly affect the detectability of target elements. Again, this is an advantage over the other models.

The size and complexity of the scenario can be quite extensive if desired. The physical size of the scenario area is limited to 30 km by 30 km which is not a significant restriction. Otherwise, the scenario is limited by model code array dimensions. Time slices may be necessary for extremely dynamic scenarios. These array limitations are shown in Appendix O. They could be modified if necessary.

There are a few deficiencies in the scenario methodology. First, the model was originally designed for a low-intensity (South East Asia) battle environment — hence, the modeling of the unattended ground sensor (UGS). The modeling of UGSs probably was responsible for the high degree of detail available in the model. However, UGS systems are not as important in the mid-intensity TNF environment. The extremely detailed battlefield environment can be a disadvantage when not required. It tends to complicate a problem common to all large simulations—that of the large-input-data requirements. It is difficult to assess whether the input requirements would be a greater problem for STANO-SAM when compared to the other two large models (SCREEN and CAMWTH). However, the extreme complexity and interdependence among STANO-SAM data sets would lead one to believe that the data problem could be far more severe. In any case, developing a new, large-scale

scenario using STANO-SAM should not be attempted by a first-time user of this model. (A similar statement could probably be made about the SCREEN model.)

Excluding the rather substantial input requirements, the scenario methodology has two other deficiencies which require modification of the code. First, there is no latitude on target composition. A target must be a group of one of the allowed elements. Besides being limited as to the type of allowed elements, the requirement of allowing only one element-type per target is very restrictive. A modification is required to allow more (or, at least, different) element types and more than one element type per target. Second, the user has no control over the detectability of the target elements (other than by changing sensor parameters). The physical characteristics used by the sensor models are specified in the code and not via input. This must be changed if evaluations are to be made on BLUE targets.

Given these modifications and accepting the possibly severe input requirement, STANO-SAM must be considered as having the most comprehensive "scenario history" of the military action of all the models under consideration.

(3) **Surveillance Methodology.** STANO-SAM has three possible sensor-deployment classes: unattended ground sensor (UGS); attended, stationary-scan sensors; and attended, moving sensors. The sensors which fall into each of these categories are listed in Appendix P.

All of the bookkeeping necessary to determine whether or not a given sensor is to investigate a given target and the conditions (LOS, environmental, and sensor parameters) of the interaction are handled in the PRERUN program. If PRERUN determines that an interaction will occur, it creates a sensor-interrogate event which will later be processed by the MSM.

The individual sensors in the scenario are specifically deployed on the map grid by input to PRERUN (planner input). The details of the deployment depend on the sensor category (UGS, stationary scan, or moving sensor). For the fixed-sensor classes, individual sensors are given specific locations on the map grid and specific times of emplacement. PRERUN routines compute location errors for positioning and probabilistic variations in time of emplacement/cease operations. In addition, times of possible sensor failures, false alarms, and repositioning are computed. Similar information is computed for monitors and data links of UGS arrays. In this manner, ground-truth positions and times of operation are computed for all fixed sensors. Ground-truth information must also be computed for moving sensors. For these sensors, this data will be a continuous function of time. Possible moving-sensor mission aborts and navigation errors are treated probabilistically.

Once the ground truth is known for each sensor, the area of sensor coverage (FOV) is determined as a function of time. There are two possible geometrical areas of sensor coverage — a circular arc area (includes arcs from 0° to a full circle) and a rectangular area. Moving sensors use a moving rectangle. Maximum and minimum ranges and FOV orientation are also considered, allowing sensor FOVs to be the regions between arcs of two concentric circles or FOVs to the side or front of the moving sensor.

LOS considerations are deterministically computed for large-scale terrain effects. That is, using the digital terrain information and foliage height in the region, the LOS is checked to determine if there is an obstacle between target and sensor. Smaller scale terrain and foliage effects in the immediate target vicinity are computed probabilistically. From these macro and micro effects, the LOS decision is made in PRERUN before any sensor-interrogate command is generated.

Most of the model bookkeeping to determine ground-truth information is processed by PRERUN steps 1 through 6 (paragraph d(1)). False alarms are generated by step 7. Sensor-target processing is done in PRERUN steps 8 through 10 and in the MSM.

In PRERUN step 8, all geometrical and temporal overlaps between targets and a given sensor's FOV are considered. Each sensor is considered individually in turn. The model completes all target interactions with one sensor before considering the next sensor (i.e., sensor processing is not chronological within PRERUN step 7). When a target enters the FOV of a sensor, a possible interaction is initiated. Additional interactions are initiated at periodic time intervals as long as the target remains within the FOV. New interactions are initiated by PRERUN when the target array within the FOV changes (i.e., a different target enters or leaves the FOV). Conditions of the interaction (environmental and sensor parameters, range, etc) are specified according to the ground-truth conditions at the time of the interaction. Once all geometrical and temporal interactions are computed for all sensors, PRERUN step 9 checks LOS for each interaction (unless sensor is LOS independent). Any interaction which is LOS blocked is simply dropped from consideration. PRERUN step 10 time orders all interactions and generates sensor-interrogate events to be used later by the MSM. The target-detection decision is made in the MSM. The output of the MSM is a time-ordered history of all sensor-detection reports.

The MSM-detection decision is determined differently for each sensor submodel. For most unclassifying UGSs, the detection decision is based on whether or not the target signal is above a threshold and various time criteria (i.e., various sensor logic allows detections to occur in only specified periods after previous detection reports). Other sensor submodels compute the probability of detecting at

least one element in the target. The detection decision is then based on a random-number draw compared to this probability.

In general, target detection implies only that the target's presence has been discovered; there is no decision as to whether the target elements can be identified as to type and number. However, the model does include classifying logic in various unattended ground sensors which simulates the sorting of target signals into different channels in an attempt to identify the number and type of target elements. The ability of attended sensors to classify target elements must be simulated by post-MSM programs.

The report of sensor histories generated by MSM includes all detections. Detection and re-detection of the same target by the same sensor must be sorted by post-MSM programs.

There are several sensor classes not explicitly modeled in STANO-SAM. These include all SIGINT sensors and various specialty sensors such as sound and flash ranging. It should be possible to mimic the technical aspects of these sensors with the general sensor type "Blackbox." However, it would also be necessary to include, in the scenario history, the events associated with these sensor-detecting targets (radio or radar emissions and artillery fire). These activities should be included in the processing sequence which generates the sensor-interrogate events.

(4) One-on-One Sensor Models. Although most of the technical details of sensor operation reside in the MSM sensor submodels, there are a set of technical background models in the PRERUN program (Step 7). The background routines are responsible for those factors affecting sensor performance which remain constant over a relatively long period of time and are generally independent of target activity. These factors are primarily environmental in nature such as background noise and radiance levels. The parameters are supplied as part of the input to the sensor models in the MSM and are also used internally by PRERUN to generate false-alarm events.

The core of the STANO-SAM model system is the set of technical sensor subroutines in the MSM. There is a submodel for all the sensor systems listed in Appendix C.

In addition to the specific sensor systems modeled, there is the sensor "Blackbox." While all of the other generic sensor models simulate the physical interactions related to specific devices and target-element properties, the "Blackbox" sensor model does not explicitly recognize any particular attributes of sensors or target-element signatures. It is a model of hypothetical sensor performance. This performance is specified in terms of a detection probability versus range curve for

specific, target-element types. It is more than a look-up table since the curve can be modified by the program depending on environmental and target-element attributes. The curve itself is of a general form defined by two parameters. The sensor performance can be made dependent on various factors by influencing the dependency of the two curve-defining parameters. This particular sensor can be used to handle sensor types not explicitly modeled in STANO-SAM.

The remaining sensor submodels are all simulations of one or more specific devices. The three generic classes of attended sensors are simulations of devices working three sensor bands. The image sensor includes all devices which image in the visible and near infrared regions (unaided visual, binocular, passive night vision devices, and TV). The thermal sensor simulates all devices which image in the thermal IR (FLIR, line scanners, and hand-held thermal viewer). The radar devices are ground based on airborne MTI systems or CB/CM radars. Different types or different models of sensors within a generic type can be simulated by specifying different sensor parameter sets.

The sensor subroutines require technical system data on the sensors, physical target element data, technical environmental data, and a few operational variables. This information is supplied from input PRERUN events or is resident in MSM data arrays. Some of the information in the MSM data arrays was also supplied by PRERUN which had computed the data from user input. However, an important subset of the data is model-designer input which is resident in the code. Technical target element data (size, temperature, reflectance, radar cross-section, permeability, noise level, seismic source strength, etc) are an important class of designer data. This data cannot be affected by user input and can be modified only by changing data statements in the source code.

The sensor subroutines use the physical information to perform the detailed sensor physics calculations necessary to decide whether or not the target is detected by a given sensor. (If one element is detected, the target is considered detected.) The output of the unclassifying UGS subroutines is simply the detection decision. The classifying UGSs yield a detection decision, a target classification (personnel, wheeled vehicle, etc), and a multiplicity index (single or multiple elements). Although a given target may contain only one element type, more than one target may be in the sensor FOV. Hence, the sensor could list more than one target type in any given report. These multiple-target reports must be sorted by post-MSM programs. The possibility of failure to classify or of incorrect classification is considered probabilistically. In that case, the output of the submodels would contain only the incorrect classification or simple detection. The attended sensor subroutines output the detection decision, the probability of detection, and the technical sensor performance parameter on which the decision is based (signal-to-clutter ratio for radars;

contrast and number of resolution elements for thermal sensors; and the ratio of subtended angle to the minimum resolvable angle for image).

With the exception of the image model, the sensor subroutines were specifically designed by CALSPAN for use in the STANO-SAM I model. The image model in SAM I was an extension and modification of a Research Analysis Corporation model of night vision devices used for the CARMONNETTE model. Additional sensor models (EMID, "Blackbox," and CB/CM radar) have been added, and the original versions of the SAM I routines have been revised and updated through the most recent version of STANO-SAM. In this respect, the one-on-one models are quite current.

When combined with the environmental and operational considerations in PRERUN, the technical aspects of the computations are excellent and obviously highly detailed. There are, of course, other one-on-one models which may have more technical detail concerning specific sensors. However, none of these models is placed in an operational environment. Of the four models under consideration, STANO-SAM must be judged as having the most current, more highly detailed set of one-on-one models.

(5) Target Definition and Total Target Detection. The output of the MSM consists solely of sensor-detection reports. For the unclassifying UGSs and the attended sensors, the reports do not indicate the type or number of target elements in the target. The classifying UGSs do report the element and whether the target contains single or multiple elements if possible. There is no attempt in any of the sensor reports to simulate inferences which may be made concerning the type of target unit to which the elements belong. If these inferences are to be derived, they must be done by some post-MSM processing program.

Currently, there are two post-MSM subprograms which attempt to carry the analysis beyond the raw-sensor reports. (In addition, there is the Tactical Communications Model, but this post-MSM program does not attempt to further analyze the raw-sensor reports.) The two analysis programs are the Unattended Sensor Model (ANALYZE) and the Attended Sensor Model.

ANALYZE is a rather sophisticated program which simulates the analysis done by human monitors of UGS arrays. These monitors interpret the activation history of UGS arrays and try to discriminate real targets from false targets and false alarms. Succeeding in this, the monitors also attempt to identify the type of target elements and to determine their course, speed, and time of arrival at future points (firetraps). Realistically, the monitors may use nonsensor information (e.g., knowledge of the movements of their own forces, natural phenomena, and perhaps

reports from other sensor types). However, ANALYZE simulates only that portion of the monitor's analysis which uses the sensor-activation history along specific paths. The output of ANALYZE consists of the perceived, target-element type and speed and times to set firetraps.

The Attended Sensor subprogram simulates attended, sensor-operator reports. The primary emphasis is on radar-type sensors. The model uses the signal-to-clutter ratio (contained in MSMOUT) to determine the level of discrimination for each radar, target-detect event. The results of the model indicate whether the operator will be able to discriminate between vehicular and personnel targets and whether he will be able to determine if there is more than one target element. In a sense, the Attended Sensor model is a recognition submodel. For the other two attended generic types (thermal and image), no significant computational decisions are performed, perhaps corresponding to an assumption that the operators of these sensors could easily perform the discrimination tasks. The output is in the form of simulated operator messages.

Currently, there are no other programs to perform further target definition or to consider multiple-sensor reports. Hence, it can be stated that the STANO-SAM system has no effective methodology to perform these two functions. It may be possible to use STANO-SAM in a war-game atmosphere similar to the manner SCREEN was designed to be used. That is, have teams of human analysts handle opposite sides of the scenario (RED and BLUE). If this were desired, the binary output of the MSM could be used as input to a new, post-processing program which would generate time-ordered reports suitable for use by the analysts in the war game. The present tactical communications submodel could also be incorporated in this process. Another possibility would be to modify the existing target-definition methodology of the CAMWTH model to use MSMOUT. This would allow the determination of perceived unit identities and probabilities.

Although the existence of the binary file, MSMOUT, makes both of the above possibilities more feasible, either of the options would probably require a rather extensive programming effort. However, if this model is to be used for the TNF, some methodology to perform the target definitions and total-target-detection functions must be devised. The present output of the MSM is clearly insufficient for TNF.

(6) Other Features of STANO-SAM. STANO-SAM has no other features beyond the simulation of STANO systems.

(7) STANO-SAM Summary. STANO-SAM is a system of models which incorporates technical sensor assessment models in a very detailed operational

environment. There are two major subprograms in the system — PRERUN which is responsible for the bulk of the operational aspects, and the Main Simulation Model which contains the one-on-one technical sensor models.

In short, STANO-SAM includes technical sensor models in a war game. In this sense, it is very similar to the SCREEN model. Although the programming approach is quite different, these two models try to accomplish essentially the same goal. Both of these models are strong on scenario and technical detail but weak on target definition and total-target detection. Both would require significant modifications and/or a man-in-the-loop procedure to perform these functions. They are both stochastic simulations which means that model results are only probable outcomes of the exercise and not expressed in probabilities. Both models require the modeling of additional sensor types although the use of the "Blackbox" sensor in STANO-SAM would probably suffice for most of the missing sensors. With the exception of additional sensor types, the solutions to the above deficiencies would probably best be implemented by post-processing programs using the current or slightly modified output of these programs.

Assuming that the major deficiencies (target definition, different sensor types, etc) can be resolved in some manner, the following modifications to the basic STANO-SAM model would be desirable:

- The physical characteristics of target elements should be input rather than having this data reside in code.
- The attended sensor models should contain recognition submodels and should output the probability of detecting and recognizing individual target elements.
- More than one element type should be allowed in individual model targets.

Overall, STANO-SAM is an exceptionally well-programmed simulation with many attractive features. STANO-SAM is superior or equal to any of the other models in many areas:

- STANO-SAM has a very detailed and realistic treatment of the scenario history. Nearly all of the scenario bookkeeping is handled by the model.
- It has a deterministic treatment of large-scale terrain effects for line of sight. Although STANO-SAM is cumbersome, this is a distinct advantage over probabilistic LOS.

- The one-on-one sensor subroutines have more technical detail than any of the other models.
- Time is handled in a deterministic and straightforward manner.
- There is detailed modeling of environmental factors (weather, background, cultural interference, etc). This modeling is also deterministic in its approach.

Clearly, if one required that the overall TNF model consider detailed technical one-on-one models, then one must use some modification of either STANO-SAM or SCREEN. Before using either, one must be aware that very detailed planning, problem definition, scenario code design, and possible source-code modification are required. Since both models have similar strengths and weaknesses, the choice between the two is primarily a trade-off of required level of detail, flexibility, and ease of use. Clearly, STANO-SAM is superior in the amount of detail modeled, while SCREEN is probably more flexible, especially in the area of target modeling. It is not clear which of the two models is easier to use.

STANO-SAM is far larger and more complex, but this may well be offset by the higher level of detail in the scenario methodology and the greater amount of internal bookkeeping. Much of STANO-SAM's complexity is due to the modeling of UGSs which probably can be ignored for TNF purposes.

The fact remains that neither STANO-SAM nor SCREEN is very easy to use and that neither is sufficient by itself. This implies a great deal of input coding and probably some additional post-processing programs. Hence, the question of whether to use STANO-SAM or SCREEN is probably best answered by whoever will actually do the code modifications and final computer analysis. Whichever model is the most familiar to that user should be the one selected.

4. Overall Model Comparisons. Each model was investigated as to how well it performed the following functions:

- Target Scenario Methodology.
- Sensor Scenario Methodology.
- One-on-one Sensor Methodology.
- Target Definition Methodology.
- Total Target Detection Methodology.

None of the models adequately performs all of the above functions. Depending on the required detail of the analysis, either modification of the models or human analysis external to the models is required to alleviate the deficiencies. The

Table, page 64, illustrates the relative capabilities of each model under each of the five functions. Also shown is a comparison of the complexity of the models and other uses for which the models were designed. A numerical ranking from 1 (best of the four) to 4 (worst of the four) is included in each table entry. This ranking is an estimate of how well the given model performs the given function when compared to the other three models.

In addition to the comparison of model capabilities in these areas, an overview of the general model approach and the objective of each of these models were included in the previous chapters. Using the Table, page 64, and the model overviews, one can draw the following conclusions concerning each model.

a. SCREEN/CRESS is a large, rather complex stochastic simulation of a reconnaissance system in a war-game approach. The main objective of the model is to evaluate the amount of information gained by a surveillance system in an operational environment. It includes an adequate set of technical one-on-one models for most sensor systems. It has an excellent and flexible representation of the target scenario in both the airborne and ground versions of the model. Separate codes for airborne and ground sensor systems are cumbersome and require that results be combined external to the codes. The model is difficult to use and has no formal one-on-many or many-on-many methodologies. This makes it essentially useless for the TNF unless new post-processing programs are coded to do these tasks or some formalized man-in-the-loop procedure is used. Either of these approaches can be used to generate detected-target lists but will probably be very time consuming.

b. The CAMWTH target-acquisition methodology represents only a portion of the present model code. The rest of the model performs fire planning and damage assessment. It is an expected-value approach which uses input tables for the one-on-one methodology for detecting elements. The representation of the target and sensor scenario is adequate but lacks the flexibility of SCREEN in the specification of target composition and the very detailed internal bookkeeping and environmental considerations of STANO-SAM. It is the only one of the four models which has a formal, target-definition methodology. This methodology could be rather easily modified and used very effectively for the TNF/S program. The many-on-many methodology is weak and should be changed before it is used.

c. The SAI target-acquisition model is a straightforward, many-on-many model. With the exception of the COMINT submodel, one-on-many detection data must be supplied to the model via input. This may be difficult because most experimental data is of the form of one sensor against one target element. The target scenario consists only of the target units without internal composition. It is the only model of the four which specifies the sensor scenario in terms of sensor assets as

Model Capabilities

Model	Target Scenario Methodology	Sensor Scenario Methodology	One-on-One Methodology	Target Definition Methodology	Total Target Acquisition Methodology	Complexity	Other Model Features
SCREEN/CRESS	(1) Very flexible and quite detailed. Allows arbitrary numbers and types of elements in targets. Biggest drawback is moving targets which must be modeled as separate targets along path of motion.	(2) Adequate although different models exist for airborne, ground, and SIGINT sensors.	(2) Adequate technical models with good flexibility and 3 levels of discrimination in sensor reports.	(3) Essentially no coded methodology. Model designed for closed war game (man-in-the-loop).	(3) Again, no formal methodology. War-game approach can be used to obtain detected-target lists.	(3) Very complex requiring much input and a great deal of prior planning.	None
CAMWTH	(3) Adequate representation of targets although somewhat limited in allowed target composition. Again, moving targets must be modeled as separate targets along path of motion.	(3) Expected value calculations using specific sensor deployments. Presently, a problem with temporal considerations.	(3) Look-up Tables	(1) Only model with formal coded methodology. With some modifications, can be quite flexible and useful.	(2) Generates detected-targets list by choosing one of the possible sensor reports. No real combination of multiple-sensor reports.	(2) Target acquisition module of the model is relatively simple but still requires a large amount of input.	Can evaluate battle-field casualties resulting from detections.
SAI	(4) Targets are only defined as TO&E units with no internal composition. Moving targets not handled well.	(4) Only model with sensor scenario defined in terms of specific deployments. Relatively simple.	(4) None	(2) One-on-many data supplied as input with the exception of COMINT submodel.	(1) Best methodology of the 4 models. Essentially, a combined-acquisition probability assuming sensor independence.	(1) Quite simple in the formulation but still requires a good deal of input.	Output of target acquisition can be used as input to weapons-allocation and damage-calculation modules.
STANO-SAM	(2) Very detailed representation handling all scenario movement with internal model bookkeeping. Severely restricted as to allowed-target composition.	(1) Excellent representation of sensor deployment. Handles nearly all of the operational bookkeeping internal to the model.	(1) Very detailed current technical sensor models. Needs modification allowing different levels of discrimination for sensor reports.	(4) Essentially no coded methodology. Model can be used in closed war game similar to SCREEN.	(4) Essentially no formal methodology. Again, may be used in a war-game approach.	(4) Extremely complex using many interrelated input data sets.	None

opposed to the specific deployment of individual sensors. The types of sensors modeled are presently too general in their nature. The model computes the total-acquisition probability on the assumption that each sensor detects independently (i.e., there is no sensor synergism). The advantages of the model are simplicity, quick execution times, and the fact that the output is in a form directly usable by the TNF (unit-acquisition probabilities and detected-target lists). The disadvantages are that one-on-many data is required and the lack of sensor scenario detail.

d. STAN-SAM IV is a very complex stochastic simulation of sensor systems operating in a battlefield environment. The overall objective is very similar to that of the SCREEN model. The model has extremely detailed internal bookkeeping of both target and sensor scenario, environmental effects, and line-of-sight considerations. It also possesses the best set of technical one-on-one submodels of all the models. However, the targets and target elements allowed in the model are somewhat restrictive, and the attended sensor subroutines do not have recognition submodels. STAN-SAM, like SCREEN, can be difficult to use and does not possess any formal one-on-many or many-on-many methodologies. Hence, it is probably not useful to the TNF in its present form. Like SCREEN, some methodology (either new post-processing programs or man-in-the-loop war game) must be devised to perform the target-definition and total-target-acquisition functions.

5. **Alternatives for the TNF/S Study.** Before proceeding to a detailed discussion of three possible alternatives, a significant point should be reemphasized. All three of the proposed approaches attempt to compute information of the detectability of target units. The combined-acquisition probabilities, target-unit identities, and estimated target sizes are important considerations for unit detectability. With the exception of the first alternative, the proposed approaches can still accommodate those studies for which detection of target elements is sufficient.

a. **Alternative 1: Use the SAI Model as it is presently configured.**

It has been assumed from the beginning that the target-acquisition methodology to be used for the TNF should be able to supply the total-acquisition probability for each target in the BLUE array. At the very least, a time-ordered list of detected targets would be necessary. Hence, it must be assumed that a many-on-many methodology has the highest priority. Of all the models, only the SAI model currently has a formal many-on-many methodology with any validity. The model in its present form produces both total-acquisition probabilities and a list of detected targets. If only one of the models is to be used and there is no desire to expend any effort on code modification or development, then the SAI model must be used. Very little code modification would be necessary beyond the possible incorporation of

additional sensor types. The model is essentially ready for use and requires only the scenario coding and the input data.

However, users must be able to acquire the necessary one-on-many data. This data is not readily available; therefore, substantial effort external to the model is required. In addition, since one is sacrificing much scenario detail, caution is required for dynamic scenarios. These are the prices one must pay for model simplicity.

b. **Alternative 2: Use a modified CAMWTH model as a base.** If other model capabilities such as more detailed one-on-many or one-on-one methodologies are desired, it would be better to use a model other than the SAI as a base. The SAI methodology for multiple sensors can be adapted to the other models if desired. This would be preferable to either modifying the SAI model or using the other more complex model to generate data to be used in the SAI model.

The CAMWTH target-acquisition module represents the next level in complexity and detail. This model considers a more detailed target and scenario than the SAI model in that it has a scenario time base, internal composition of target units, and specified deployment of sensors. In addition, it has both one-on-one and one-on-many methodologies. However, it will be necessary to devote some initial effort into code modification. Given the code development, the CAMWTH target acquisition module can be made into a model well suited to the needs of the TNF study. In addition, the CAMWTH model assesses casualties which is necessary to determine the worth of counter-target-acquisition measures.

In paragraph b(8) (CAMWTH Summary), the following eight modifications were provided:

- (1) Improve many-on-many methodology.
- (2) Include the effects of time in the interactions between target and sensor pairs.
- (3) Give special considerations for certain element-sensor pairs (e.g., radios - RDF sensors.)
- (4) Include target location in the identification methodology.
- (5) Increase the number of allowed-element types.
- (6) Change target and sensor-processing sequence.
- (7) Allow different levels of discrimination in element detection.
- (8) Incorporate proper many-on-many results into the damage calculations (does not affect target-acquisition module).

Exact details and feasibility of implementation of all the above suggestions cannot be precisely stated at this time. It is quite possible that after more detailed consideration, some of the code changes may be too difficult to implement or not desirable enough to merit further effort. It is not necessary to make all of the modifications for the model to be usable for the TNF. However, the first modification is mandatory. The present method by which a detected-target list is generated is not acceptable. There are several approaches which may prove to be adequate, and the contractor is currently making some of these modifications. A brief description of one possible approach to the computation of the target's total-acquisition probability is outlined in the following paragraphs.

Consider an individual sensor i of type j which is investigating a given target. The present, target-acquisition module computes the probability of detection, the expected target identity, the expected size of the target, the time of detection, and the expected location error. In the detection calculation, all the relevant probabilities are considered at the time of target-sensor interaction. Each sensor report is made independent of all others.

To modify this procedure for multiple-sensor considerations, compute and store for each of the individual target-sensor interactions the detection probability, the associated probabilities for each of the possible sizes, errors, time factors, and conditions of the interaction. There are certain conditional probabilities relating to the target which are necessary for detection (e.g., probability that the target will emit the signature that the target is masked by terrain or foliage in the target vicinity). These conditional probabilities may or may not be functions of time. Since these probabilities are related to the target alone, they affect all appropriate sensors in the same manner. They, therefore, should be excluded from the independent-sensor-detection calculations.

Hence, for each sensor which has a nonzero probability of detecting the given target, one has stored:

PD_i = probability of detection given that conditional events have occurred and infinite search time.

$\left. \begin{matrix} P_{1i} \\ P_{2i} \\ P_{3i} \end{matrix} \right\}$ relative probabilities that the target would be sized in each of the possible categories given detection.

t_{Ri} = report time delay.

t_{Si} = search time factor.

E_i = expected location error.

t_{1i} = sensor start time.
 t_{2i} = sensor end time.
 I_i = perceived target identity.

To combine probabilities across sensors, it is necessary to have different categories of sensors. These categories are simply groupings of sensors which require similar conditional events. (Certainly, all sensors of the same type j will be in the same category.) Label these categories by the index k ($k=1, NV$). Associated with each of these categories are the required conditional probabilities $P_{\text{cond}}(k,t)$.

For each individual sensor i , there is the detection probability

$$PD(i,t) = PD_i \left\{ 1 - \exp \left[- \frac{t_m - t_{Ri} - t_{1i}}{t_f} \right] \right\}$$

$$t_m = \max [t, t_{2i}]$$

$$t_m - t_{Ri} - t_{1i} \geq 0$$

$$PD(i,t) = 0 \text{ for } t_m - t_{Ri} - t_{1i} < 0$$

Combining across all sensors of given category

$$PD(k,t) = \left\{ 1 - \prod_{i \in \{k\}} [1 - PD(i,t)] \right\} P_{\text{cond}}(k,t).$$

Note that $P(k,t)$ may well depend on periods of individual, target-sensor overlap since it basically is the probability that the sensors will have been given an opportunity to investigate the target.

The total acquisition probability is then

$$PD(t) = 1 - \prod_k \{ 1 - PD(k,t) \}.$$

For the size of the target, assume that there is some user specified preference as to what size unit would most likely be deployed. For example, say the sizes are 1 = firing section, 2 = platoon, and 3 = battery. Also, assume that the most likely deployment is battery (3) with firing section (1) next most likely and platoon (2) least likely.

Therefore, let

$$P_3(i,t) = PD(i,t) P_{3i}.$$

Combine again over $P_3(t)$ in a manner similar to the detection probability.

Then, let

$$P_1(i,t) = PD(i,t) P_{1i}.$$

Combine again as before to obtain $\bar{P}_1(t)$. Now, let the total probability for size 1 be

$$P_1(t) = \bar{P}_1(t) (1 - P_3(t));$$

similarly, for $P_2(t)$,

$$P_2(t) = \bar{P}_2(t) (1 - P_1(t)) (1 - P_3(t)).$$

One can then normalize the three probabilities by simply dividing by $PD(t)$.

A similar procedure can be used for target identification. Simply set up a priority of target identities perhaps based on target value and level of sensor discrimination. Then, compute the probability that the highest priority identity will be chosen. This is simply the combined probability (in the same manner as the detection probability combinations) of all sensors which identified the target as this highest priority probability. Call this $P_{11}(t)$. Then, compute (in a similar manner) the probability for the next highest priority identity ($\bar{P}_{12}(t)$). However, one must reduce this by $P_{12}(t) = \bar{P}_{12}(t) (1 - P_{11}(t))$; similarly,

$$P_{IN}(t) = \bar{P}_{IN}(t) [1 - P_{11}(t)] [1 - P_{12}(t)] \dots [1 - P_{I(N-1)}(t)].$$

Again, these probabilities can be normalized by simply dividing by $PD(t)$. The location error can be handled by simply assigning some confidence level to each sensor type (j). That is, assign some value (say between 0 and 1.0) for each sensor type reflecting the preference of an intelligence analyst to choose one sensor report over another for specification of an aimpoint. For example, suppose two reports reach an analyst — one from an RDF sensor with poor-location error and one from a patrol

with good-location error. The analyst is likely to choose the patrol report for specification of an aimpoint.

Let us assume that the user has specified the preferences for each sensor type ($Pf(j)$). The location error is then

$$E(t) = \frac{\sum_{i=1}^{NS} PD(i,t) Pf(j) E_i}{\sum_{i=1}^{NS} PD(i,t) Pf(j)} .$$

Outlined above is a rather complicated methodology for multiple-sensor considerations. There are obviously many details to be resolved by the modeler. The above approach attempted to maintain the expected-value nature of the CAMWTH methodology and also to rigorously maintain the time variable. At some stage of the calculation, it may be necessary to abandon both objectives. One may have to average over the temporal variable to simplify calculations; otherwise, the computations may become too cumbersome even for a computer. This obviously requires a compromise between accuracy and practicality. It is believed that time considerations are extremely important in the acquisition process. Hence, it would be unwise to remove the temporal considerations too early in the computation. Given time averages, the computations become simpler and the expected-value approach could be maintained.

c. **Alternative 3: Use one of the large simulation models (SCREEN or STANO-SAM) as the base model.**

If one needs one-on-one data, it is possible to use a good one-on-one model (e.g., MARSAM II) to supply the data. However, most one-on-one models are not in operational environments, and the user must be careful to assure that the inputs to the one-on-one model reflect the conditions he wants to consider in the larger scenario.

If one desires to have technical one-on-one considerations as part of the overall model without external manipulations, then it is necessary to use one of the two large simulation models — SCREEN or STANO-SAM. These two models have the necessary scenario detail to supply the proper data to their one-on-one technical models. This level of detail is absent in either CAMWTH or SAI. SCREEN is probably better suited to large scenarios and has greater flexibility in internal-target composition; whereas, STANO-SAM has greater internal-scenario bookkeeping and more current one-on-one models with a higher level of technical detail. Both models require

an expanded set of sensor types and several internal modifications which can be significant, coding-development efforts. The required internal modifications are listed in paragraph a(7), (SCREEN Summary), or in paragraph d(7), (STANO-SAM Summary), depending on which model one chooses as a base.

The major deficiencies of both models are the lack of formal one-on-many and many-on-many methodologies. A war-game, man-in-the-loop approach can supply these methodologies. (SCREEN is probably better suited for this approach.) However, the man-in-the-loop is *not* recommended since it lacks repeatability and probably requires as much pre-planning and analysis effort as developing post-processing codes. Hence, it is necessary to develop a post-processor to do target definition and to compute total-acquisition probability. To implement this alternative:

- (1) Choose between SCREEN and STANO-SAM. Each has its relative strengths and weaknesses in certain areas. The person or persons actually responsible for the code modifications should have significant input into this decision.

- (2) Implement the modifications suggested in the summary for the chosen model.

- (3) Develop a post-processor which will sort the present model output so that all reports on the same target are concurrent. Be sure that element-detection probabilities are retained. Ignore the stochastic decisions made by the model as to whether or not the target is detected and the predicted number of detected elements. Work with the element probabilities.

- (4) Modify the CAMWTH one-on-many methodology to use newly sorted output. The output of this module should be in the form specified in Alternative 2.

- (5) Develop a new many-on-many methodology to use the output of the above stage. The methodology outlined in Alternative 2 should suffice. It may even be easier to implement here since many of the conditional probabilities could probably be handled in the base model because of its scenario detail. The output should be tailored to the needs of the TNF study.

If all of the above suggestions are implemented, one would undoubtedly have an extremely comprehensive and useful model. However, one must be cautioned that such an undertaking would entail substantial code development and would result in a model with substantial input requirements.

III. CONCLUSIONS

6. Conclusions. It is concluded that:

a. There are three possible approaches concerning model usage for the target-acquisition requirements of the Theater Nuclear Force survivability and vulnerability assessments:

(1) Use the SAI model as it is now configured or with minor modifications.

(2) Use the CAMWTH model as a base. Change the code as appropriate to correct deficiencies, especially for a better many-on-many methodology.

(3) Use one of the large simulation models (SCREEN or STANOSAM) as the base model. Develop post-processing routines to incorporate modified CAMWTH target-definition methodology and an appropriate many-on-many methodology.

b. The first of the three alternatives would be the easiest and quickest to implement and use but would yield results with a low level of confidence. This approach also requires one-on-many input data of which little exists. The third alternative requires the most effort but would be the most sophisticated with the most validity (hopefully). The middle alternative is a compromise between the two.

c. Alternative (2) was chosen as best suiting the needs of the TNF/S study with regard to what target-acquisition data is presently available or could reasonably be made available from planned tests for model input, timeliness or TA model availability, resource restrictions for model modification and testing, and quality of anticipated results.

d. Alternative (1) is not acceptable due to the lack of and difficulty of obtaining one-on-many sensor/target data, lack of a target definition routine, lack of a one-on-one sensor model routine, and overall model simplicity with little confidence in the method of calculating total target-acquisition probability.

e. Alternative (3) would be the best approach with regard to quality of results (assuming adequate and extensive revisions could be made) but does not meet the resource and time requirements of the TNF/S program.

APPENDIX A

SCREEN INPUT DATA

1. SCENARIO DATA

a. Target Characteristics

- (1) time and position coordinates
- (2) object types
- (3) background data
- (4) antiaircraft capabilities

b. Atmospheric Data

2. PHYSICAL CHARACTERISTICS

a. Object Data (composition, size, etc)

b. Physical properties of materials

c. Failure rates of sensors, aircraft, navigation systems, observation post/patrol (OP/Patrol) attrition

3. FLIGHT PARAMETERS (SCREEN-AIR)

a. Platform type

b. Takeoff and landing time and position coordinates

c. Sensors aboard

d. Navigation system

e. Speed and altitude

f. Reconnaissance-Surveillance (RS) areas searched (flight pattern)

4. OBSERVATION POST AND PATROL DATA (SCREEN-GROUND)

a. Map coordinates and OP height

b. Times operating

c. Communications link

d. Sensors

e. Line of sight probability to *each* target

APPENDIX B

PHYSICAL DATA REQUIREMENTS

1. OBJECT DATA

- a. Size (length, width, height)
- b. Shape factor
- c. Composition (types of materials of which it is constructed)
- d. Pattern of arrangement of material types

2. BACKGROUND DATA

- a. Composition
- b. Pattern of material arrangement

3. MATERIAL PARAMETERS

- a. Visual region reflectivity
 - b. IR region reflectivity
 - c. IR emissivity
 - d. Radar region reflectivity
 - e. Average surface temperature
-

APPENDIX C

TYPES OF AERIAL AND GROUND SENSORS MODELED

AERIAL

1. Camera
 - Vertical Frame
 - Side Oblique
 - Panoramic
2. IR Line Scanner
3. Side Looking Airborne Radar
4. Visual
5. Laser Illuminator
6. Low Light Level TV
7. Reconnaissance by Fire

GROUND

1. Thermal IR
 2. Ground Surveillance Radar
 3. Passive Night Vision Device
 4. TV
 5. Visual
 6. Laser Scanner
 7. IR Binocular
-

APPENDIX D

CAMERA SYSTEM PARAMETERS

-
1. Focal Length
 2. Side Oblique Depression Angle
 3. Forward Oblique Depression Angle
 4. F-Number
 5. Lens Resolution Efficiency
 6. Lens Transmission
 7. Film Resolution
 8. Filter Factor
 9. Image Motion Factors
 10. Film Width
 11. Mean Time Between Failures
-

APPENDIX E

CAMWTH INPUTS

A. MOD1 (Overlay (1, 0))

1. Target-Element Types
2. Target Data
 - a. Deployment
 - b. Composition
 - c. Times of existence
3. Sensor Data
 - a. deployment
 - b. times of existence
 - c. field of view
 - d. location CEP
4. Line of Sight Data (function of range and altitude)
5. Stylized Unit Compositions
6. Identification Vector (used by target-definition routines)
7. Detection Probabilities (This data is not card input but must reside on random-access disk. Separate program must be executed to create disk file prior to MOD1 execution.)

B1. MOD2C (Conventional Ordnance – Overlay (2, 1))

1. FEBA and Safeguarded-Area Data
 2. Fire-Unit Data
 - a. delivery systems (munitions)
 - b. coordinates
 - c. times of existence
 3. Delivery-System Data
 - a. munition data (cost, reliability, volley information)
 - b. CEP for delivered ordnance as function of range
 - c. lethality for different target types and postures
 - d. number of pieces per battery
 - e. response time
-

B2. MOD2N (Nuclear Ordnance – Overlay (2, 2))

1. FEBA and Safeguarded-Area Data
2. Fire-Unit Data
 - a. delivery systems
 - b. coordinates
 - c. times of existence
3. Delivery-System Data
 - a. warhead yield, reliability, cost, and fuzing
 - b. delivery CEP as function of range
 - c. response time
4. Damage Curves

C. MOD3 (Overlay (3, 0))

1. Selection Criteria for Fire Planning
 - a. cost
 - b. collateral damage
 - c. direct control
2. Target Array Descriptions

D. MOD4C and MOD4N (Overlay (4, 0))

Same as for MOD2C or MOD2N

APPENDIX F

CAMWTH OUTPUTS

A. MOD1

1. For each Target-Sensor Interaction and for the Real and Estimated Target (as perceived by the enemy)

- a. aimpoint coordinates
- b. location coordinates
- c. target radius
- d. time information reaches Fire Direction Center
- e. number and type of detected (or real) target elements
- f. target identity and size

2. Probability that target is detected

3. Sensor Providing Data

B. MOD2C/MOD2N

For each delivery system which can engage each Estimated Target

1. Munitions type/warhead yield
 2. Whether target was defeated (according to defeat criteria)
 3. Collateral Damage
 4. Aimpoint
 5. Number of volleys
 6. Number and type of target elements destroyed
-

C. MOD3

1. For each target engaged

- a. fire unit and delivery system engaging
- b. coverage
- c. cost
- d. number of volleys
- e. number and type of target elements expected to be destroyed

2. Summary tables for fire plan

D. MOD4C/MOD4N

Same as for MOD2C or MOD2N except that only planned fire unit engaged real targets.

E. EXECUTIVE

1. Summary tables for each target

2. Summary tables for each target-element type

APPENDIX G

SCENARIO DATA

1. Size Constraints (as model is now configured)

- a. Scenario targets – 500 maximum
- b. Stylized units – 15 user-specified types
10 general types (no user control)
3 sizes for each type
- c. Element types per target – 4 maximum
- d. Target-element types – 11 exactly

2. Scenario Target Data

- a. name and identifier
- b. stylized unit type
- c. coordinates and radius
- d. times of existence
- e. environment
- f. composition
- g. activity probabilities

Example: For a target tank company deploying in tree-line position at map grid (500, 600) at 0800 and departing at 1200.

ID: A111

Name: Armor Co.

Stylized Type: 1 (Refers to type 1 stylized unit)

Coordinates: (500, 600)

Target Radius: 100 meters

Times: Start 0800, end 1200

Environmental Code: 3 – Keyed to detection probabilities for elements in tree line.

Composition: 17 tanks, 8 support vehicles, 4 radios

Activity Probabilities: Transmitting { Keyed to detection
 Moving } probabilities for
 Shooting } specific sensors.

3. Stylized Unit Data

a. Type and Organizational size

b. Composition

Example: Types (Specific):

- | | | |
|-----------------------|-----------------|---------------------------|
| 1. Armor | 6. Supply Point | 11. Signal |
| 2. Mech Inf | 7. Tube Arty | 12. Aviation |
| 3. Infantry | 8. Missile Arty | 13. Transport |
| 4. Air Defense | 9. Headquarters | 14. Munition Supply Point |
| 5. Target Acquisition | 10. Engineer | 15. Mixed |

Sizes can be Platoon, Section, Company, Battalion, Battery, Division, Corps.

(Three for each type; can have different combination for each type.)

10 General stylized types – no user control.

4. Target Elements

Examples:

- | | | |
|--------------------|-------------|-------------------------|
| 1. Personnel | 5. Cannon | 9. Command Post Vehicle |
| 2. Tanks | 6. Missile | 10. Munitions |
| 3. APCs | 7. Radio | 11. Radar |
| 4. Support Vehicle | 8. Supplies | |
-

APPENDIX H

CAMWTH SURVEILLANCE SYSTEM INPUTS

1. Sensor Type Data (15 types)

- a. FOV type (fixed, baseline, moving)
- b. Nomenclature
- c. Ability to discriminate target elements
- d. CEP (as a function of range)
- e. Does sensor require LOS?
- f. Maximum detection range
- g. Target-Element-Detection Probabilities (as a function of range and conditions)

NOTE: Element-detection probabilities must be on random-access disk prior to program execution.

2. Individual (Scenario) Sensors – (500 maximum)

- a. Deployment (depends on FOV type)
- b. Times of existence
- c. Altitude
- d. Delay time

3. Example Sensor Types

- a. Forward Observer – Naked Eye
 - b. Forward Observer – Ground-based IR
 - c. Counter Battery Radar
 - d. Flash Ranging
 - e. Sound Ranging
 - f. Ground-Surveillance Radar
 - g. Ground-Based, Radio-Direction Finding
 - h. Ground-Based, Radar-Direction Finding
 - i. Airborne, Radio-Direction Finding
 - j. Airborne, Radar-Direction Finding
 - k. Airborne Observation Post
 - l. Airborne IR
 - m. Airborne SLAR
 - n. Satellite Reconnaissance
 - o. Airborne Photography
-

APPENDIX I

SAI MODEL INPUTS

1. General Battlefield Data and Miscellaneous (FEBA, Range Bands, Times, Monte-Carlo parameters, etc)
2. Stationary Target Data (Read separately from Mass Storage)
 - a. identifier
 - b. location
 - c. Target type (class)
3. Moving Target Data
 - a. identifier
 - b. times of movement
 - c. acquisition probability
 - d. direction, velocity, and uncertainties of direction and velocity
4. Target Class Data
 - a. class name
 - b. value
 - c. acquisition type
 - d. SIGINT type
 - e. target permanence
 - f. location error for targets of this class in all range bands
5. Sensor Data
 - a. number of sensors of various types
 - b. range bands for patrols
 - c. COMINT acquisition probability (common to all COMINT targets)
6. Detection and Classification Probabilities and Mean Times to Detect and Classify
 - a. for each sensor type versus each target class
 - b. in each range band where applicable
 - c. for each target cover and concealment mode
7. LOS Data (as function of range for each sensor type)
8. Visibility Data (for each sensor type against each acquisition type)
9. Cover and Concealment Probabilities
Probability that any given, acquisition-type target will be in the given cover and concealment mode.
10. Activity Factor for Sensor and Target-Class Pairs
11. Number-of-Looks Data for each sensor Type
 - a. Field of view for each sensor type
 - b. Maximum and minimum range for each sensor type
 - c. Survivability data for penetrating sensor

APPENDIX J

SAI MODEL OUTPUTS

(3 Tabular Listings)

1. Number of targets in Input Array and expected number acquired. Indexed by target class and range band.
 2. Average Acquisition Probability by target class and range band.
 3. Detected-Target List
 - a. target identifier
 - b. detection time
 - c. target class
 - d. target value
 - e. distance behind FEBA and range band
 - f. target-location error
-

APPENDIX K

STANO-SAM IV INPUTS

AUXILIARY SUBPROGRAMS

1. Terrain Program

- a. No card input (data statements in program set scenario bounds).
- b. TOPOCOM/ISA Terrain Tape for scenario region modeled.

2. Atmospheric Program

- a. Planner input (user can specify for any given day and hour). Data includes: solar and lunar altitude, lunar phase, amount and type of precipitation, wind speed, temperature, pressure, humidity, and cloud ceiling.
- b. Five probability data sets (for time periods in which the user has no detailed specific values; includes data by which the program will generate data left unspecified by detailed planner input).
- c. Tables for vegetation ground cover and micro-terrain.

3. Radar Contour Plot Program

- a. Output of atmospheric program.
- b. Output of terrain program.
- c. Plotting parameters.
- d. Scenario area parameters.
- e. Radar parameters (position and system parameters).

4. RF Data Link Program

- a. Output of terrain program.
- b. Output of atmospheric program.
- c. Sensor arrays linked with receiving monitor (positions).
- d. Transmitter and receiver system data.
- e. Additional environmental data.

5. Unattended Sensors Program

- a. Output of MSM (MSMOUT) (currently not properly formatted).

b. Number of sensors, monitors, and firetraps (NOTE: Would not be needed if MSMOUT could supply).

c. Planner values for decision-making thresholds.

6. Attended Sensor Program

a. Output of MSM.

b. Negligible card input (important decision-making criteria are in program data statements).

7. Tactical Communications Program

a. Output of MSM (currently not properly formatted).

b. Message precedence.

c. Communications route for each message.

d. Cycle time for message.

e. Interfering Communications Traffic rates and cycle times.

APPENDIX L

STANO-SAM IV OUTPUTS

AUXILIARY PROGRAMS

1. Terrain Program

Output is in the form of a digital elevation tape with a 100-meter (x, y) resolution.

2. Atmospheric Program

a. Printer listing of atmospheric tables as a function of time (listing can be user suppressed). Tables include information similar to that required if planner input is used. The tables also include several other physical parameters computed by mode.

b. Atmospheric and micro-terrain/foilage tables on mass storage file for use by later subprograms.

3. Radar Contour Plot Program

Outputs graphic plots for selected radars (user specified). Plots indicate which areas are visible, masked, or out of range.

4. RF Data Link Program (outputs printer listing for each transmitter path).

- a. Path losses.
 - b. Received signal-to-noise ratio.
 - c. Intermediate data used to predict path losses.
-

5. Unattended Ground Sensor Analysis Program (Printer Listing)

- a. Correlation of detection by groups of UGS sensors.
 - b. Estimated target speed.
 - c. Estimated number of target elements.
 - d. Estimated time of arrival of target at firetrap points.
-

6. Attended Sensor Analysis Program (Printout of Simulated-Operator Message)

- a. Estimated target-element type.
 - b. Target location.
 - c. Estimated number of element.
-

7. Tactical Communications Program (Printer Listing)

- a. Time history of all messages (location of the message in the communications net).
 - b. Message summary data (for each message).
 - (1) Transmitter/receiver location.
 - (2) Time entered net.
 - (3) Message length.
 - (4) Total delays.
 - c. Message delay at each link.
 - d. Operator delays.
 - e. Summaries of delay at each STANO and non-STANO link.
-

APPENDIX M

PRERUN INPUTS AND OUTPUTS

PRERUN INPUTS

(All inputs are read during Step 0)

A. Atmospheric data set (output of Atmospheric Subprogram)

B. Terrain data set (output of Terrain Subprogram)

C. Planner Card Inputs: 0-30 data sets (0 through 29)

0. Header Cards – Miscellaneous Data.

1. UGS Array Data:

- (a) ID.
- (b) Location.
- (c) Monitors and data links.
- (d) Times (emplacement, cease, replacement).
- (e) Up/down, emplacement, failure probabilities.

2. Position Errors in Stationary Sensor Emplacement (UGS and Attended Sensors).

3. Sensors (All sensors cross-indexed to various sensor arrays):

- (a) ID.
- (b) Type (generic and index to appropriate sensor-system parameters set).
- (c) Location and orientation (irrelevant for moving sensors).
- (d) Scanning and coverage type.

4. Sensor System Parameter Set:

- (a) ID of parameter set (to be cross-indexed to individual sensors).
- (b) Generic type.
- (c) Coverage data.
- (d) Technical data varies with sensor type.

5. Firetrap data.

6. Monitor Data (individual monitors).

7. Monitor Parameters (cross-indexed to individual monitors).

8. Relay Data (individual relays).

9. Relay Reliability Data (cross-indexed to individual relays).

10. Data Link Data (cross-indexed to sensor arrays and monitors).

11. Receiver/Transmitter Data (cross-indexed to monitors, relays, and sensor arrays).
12. Path (Route) Data (cross-indexed to individual sensors covering routes and moving arrays).
13. Force-Type Parameters (cross-indexed to targets):
 - (a) ID.
 - (b) Concealment mode.
 - (c) Deployment (spacing or movement formation).
 - (d) Weight type.
 - (e) Acoustic types.
14. Coverage/Scan Parameters (cross-indexed to sensors).
15. Hyperbolic Navigation System Parameters.
16. RHO THETA Navigation System Parameters.
17. Doppler Navigation System Parameters.
18. Nonspecific Navigation System Parameters.
19. Stationary Scan Sensor Arrays (Individual Sensors or individually linked arrays):
 - (a) ID.
 - (b) Generic type.
 - (c) Index to proper sensor-system parameters.
 - (d) Location.
 - (e) Times.
 - (f) Emplacement failure probabilities.
20. Moving Sensors, Targets, and False Targets:

(NOTE: Moving sensor arrays are possible targets.)

 - (a) ID.
 - (b) Types of sensors (if any).
 - (c) Navigation system.
 - (d) Mission probabilities.
21. Enemy Forces (to be treated as possible targets – see Note):
 - (a) ID.
 - (b) Type (individual, squad, tank, etc).
 - (c) Times.
 - (d) Speed and altitude.
 - (e) Route (cross-indexed to Path Data).
 - (f) Force type (cross-indexed to Force-Type Parameters).

22. Friendly targets (see Note)
Same as for data set 21 except that friendly targets could be stationary in which case location is necessary.
23. Scheduled Planner Battle Events:
- (a) Event ID.
 - (b) Location and times.
 - (c) Type (small-arms fire, artillery firing, artillery impacting, bombs impacting, etc (18 types)).
24. Random Battle Events:
- (a) Event ID.
 - (b) Probabilities of occurrence as a function of time (6-hr intervals).
25. Exclusion Areas for Random Events
(Areas in the modeled scenario area where random battle events will not occur.)
26. Firesupport Base Data Set
(Planner input to provide locations for weapons' firing battle events.)
27. Planner Specified Cultural Events (for false targets):
- (a) Event ID.
 - (b) Type (Surf, civilian vehicles or structures, animals, etc).
 - (c) Location and times.
28. Random Cultural Events:
- (a) Event ID.
 - (b) Associated expected number of occurrences (6-hr intervals).
29. Cultural X-Y Bounds

NOTE: In keeping with the terminology of this model comparison, the BLUE forces are assumed to be the friendly target array while the RED forces are assumed to be the enemy sensor array. In STANO-SAM documentation, the opposite terminology was used.

PRERUN OUTPUTS

There are printer listings to indicate results of all PRERUN steps. In addition, there is plotting capability in Step 0 and Step 5 for selected sensor, firetrap, monitor, and path data. Step 0 plots are based purely on input while Step 5 plots show perturbation of random events.

Step 10 has printer listing of all events (see below).

Main outputs of PRERUN are two disk files for use by MSM. They are:

1. **Sensor System and Environmental Parameters.**

2. **Time-Ordered Events List. Possible types are:**

- (a) **Sensor Interrogate (Target-Sensor Interaction).**
 - (b) **Sensor False Alarm.**
 - (c) **Sensor Parameter change.**
 - (d) **Sensor up/down.**
 - (e) **Monitor up/down.**
 - (f) **Data link up/down.**
 - (g) **Firetrap up/down.**
 - (h) **Emplace/cease operations.**
 - (i) **Battlefield illumination.**
 - (j) **Sensor reposition.**
-

APPENDIX N

MSM OUTPUTS

PRINTER OUTPUT

1. System parameters.
2. Time History of Sensor reports game play versus ground truth.
3. Periodic "system snapshots" of sensors and arrays active at the given stage of the simulation.
4. Effective time of emplacement or cease operations for sensors.
5. Effective times of firetrap operation.
6. Beginning and end of precipitation.

BINARY OUTPUT (MSMOUT)

Event Oriented – Event types follow with their appropriate ID:

1. Seismic (unclassifying) sensor report.
2. Acoustic (unclassifying) sensor report.
3. Magnetic (unclassifying) sensor report.
4. ARF BUOY report.
5. Passive IR (unclassifying) sensor report.
6. Radar sensor report.
7. Imaging sensor report.
8. Thermal viewer sensor report.
9. Breakwire sensor report.
10. Electromagnetic Intrusion Device (EMID) (unclassifying) sensor report.
11. "Blackbox" (unclassifying) sensor report.
21. Seismic (classifying) sensor report.
22. Acoustic (classifying) sensor report.

- 23. Magnetic (classifying) sensor report.
- 25. Passive IR (classifying) sensor report.
- 29. Conducting wire (unclassifying) sensor report.
- 30. EMID (classifying) sensor report.
- 31. "Blackbox" (classifying) sensor report.
- 50. Firetrap up/down status report.
- 51. UGS Array up/down status report.
- 100. System parameters.

NOTE: "Classifying" sensors are those with the capability of discriminating between element types and estimating the number of elements.

APPENDIX O

CURRENT STANO-SAM DIMENSION LIMITS

SENSORS	
<u>Data Set</u>	<u>Dimension</u>
UGS arrays	200
Stationary scanning sensor	300
Moving sensors ^(a)	200
Total sensors of all types	400
Different sensor types	100
Monitors	100
Monitor types	10
Relays	50
Relay types	10
Data links	500
Path data (routes) ^(b)	275

TARGET SCENARIO	
BLUE targets	100
RED targets (false targets) ^(c)	300
Total moving targets ^(a)	200
Path Data (routes) ^(b)	275

- (a) Moving sensors and targets are both stored in same array; hence, total number of moving units (targets, false targets, and sensors) must be less than 200.
- (b) Route data for targets and sensors are stored in same array.
- (c) RED targets include moving sensors and stationary RED targets in the surveillance area.

APPENDIX P

STANO-SAM SENSORS

1. Unattended Sensors

- a. Seismic (NC & C)*
- b. Acoustic (NC & C)
- c. Magnetic (NC & C)
- d. Passive IR (NC & C)
- e. Electromagnetic Intrusion Detector (NC & C)
- f. Breakwire (NC)
- g. Conducting Wire (NC)
- h. Blackbox (NC & C)
- i. Arfbuoy (NC)

2. Attended Sensors (Stationary Scan)

- a. Radar
 - (1) MTI
 - (2) CB/CM
- b. Image
 - (1) Unaided eyesight
 - (2) Binocular-aided vision
 - (3) Passive night vision device
 - (4) Low Light Level and daylight TV
- c. Thermal Sensors
 - (1) Handheld thermal viewer
 - (2) FLIR

3. Moving Sensors

Same as for stationary-scan sensors except sensor moves either along the ground with forces or in aircraft.

- * NC - non-classifying sensors.
- C - classifying sensors.

DISTRIBUTION FOR MERADCOM REPORT 2227

No. Copies	Addressee	No. Copies	Addressee
	Department of Defense		
1	Director, Technical Information Defense Advanced Research Projects Agency 1400 Wilson Blvd Arlington, VA 22209	1	Commander USAARRADCOM ATTN: DRDAR-LCF-F, Reiner Dover, NJ 07801
2	Director Defense Nuclear Agency ATTN: STNA, Col Johrsrud VLWS, Maj Chrobak Washington, DC 20305	3	Commander US Army Electronics Research and Development Command ATTN: DRSEL-WL-V, Bookbinder DRSEL-CT-R, Frost DRSEL-CT-A, Kaufman Fort Monmouth, NJ 07703
12	Defense Documentation Center Cameron Station Alexandria, VA 22314	1	Commandant US Army Command and General Staff College ATTN: ATSW-RI-L Fort Leavenworth, KS 66027
	Department of the Army		
1	Commander, HQ TRADOC ATTN: ATCD-SA, O. Wells Fort Monroe, VA 23651	6	Director US Army TRADOC Systems Analysis Activity ATTN: ATAA-TDC, LTC Hesse White Sands Missile Range, NM 88002
1	HQDA (DAMA-CSM-N) Washington, DC 20310	4	Commander US Army Combined Combat Developments Activity ATTN: ATCA-CCC-F (Col Zarembo) ATCA-CO (McElwee) ATCA-CF (Follis) ATCA-CA (Pleger) Fort Leavenworth, KS 66027
1	HQDA (DAMO-SSN) Washington, DC 20310		
1	HQDA (SAUS-OR) Washington, DC 20314	1	Commander US Army Field Artillery Center and School ATTN: ATSF-CD-DC (MAJ Renfro) Fort Sill, OK 73503
1	HQDA (DAMI-TA) Washington, DC 20314		
1	Director US Army Ballistic Research Laboratories ATTN: DRDAR-BLV, Rigotti Aberdeen Proving Ground, MD 21005	1	Commander USA Intelligence Threat Analysis Detachment ATTN: MAJ Godboldte, Bldg A Arlington Hall Station Arlington, VA 22212
2	Director US Army Materiel Systems Analysis Agency ATTN: DRXSY-S, Reches DRXSY-DS Aberdeen Proving Ground, MD 21005		

No. Copies	Addressee	No. Copies	Addressee
3	Commander US Army Concepts Analysis Agency ATTN: MOCA-WGP (LTC Jacob) MOCA-SM (M. Lawrence) MOCA-WGG (Col Hincke) 8120 Woodmont Ave Bethesda, MD 20014	1	MERADCOM Commander, DRDME-Z Technical Director, DRDME-ZT Assoc Tech Dir/R&D, DRDME-ZN Assoc Tech Dir/Engrg & Acq, DRDME-ZE Spec Asst/Matl Asmt, DRDME-ZG Spec Asst/Tech Asmt, DRDME-ZK CIRCULATE
2	Commander Harry Diamond Laboratories ATTN: DRXDO-RBA (Rosada) DRXDO-EM-2 (Gornack) 2800 Powder Mill Rd Adelphi, MD 20783	1	Chief, Ctrmine Lab, DRDME-N Chief, Engy & Wtr Res Lab, DRDME-G Chief, Elec Pwr Lab, DRDME-E Chief, Mar & Br Lab, DRDME-M Chief, Mech & Constr Eqpt Lab, DRDME-H Chief, Ctr Intrus Lab, DRDME-X Chief, Matl Tech Lab, DRDME-V Director, Product A&T Directorate, DRDME-T CIRCULATE
1	HQDA (DAMA-CSS) Washington, DC 20314	1	Cam & Topo Lab, DRDME-R Res Tech Div, DRDME-RT Tech Reports Ofc, DRDME-WP Security Ofc, DRDME-S Tech Library, DRDME-WC Plans, Programs & Ops Ofc, DRDME-U Public Affairs Ofc, DRDME-I Ofc of Chief Counsel, DRDME-L
4	Commander US Army Materiel Development and Readiness Command ATTN: DRCDE-DS (COL Jones) DRCDE-DE (Mr. Sawyer) DRCBSI (COL McDonnell) DRCBSI (Dr. Anderson) 5001 Eisenhower Ave Alexandria, VA 22333	30	Department of the Navy 1 Director, Physics Program (421) Office of Naval Research Arlington, VA 22217
1	Commander US Army Field Artillery Center and School ATTN: ATSF-CD-DC (MAJ Renfro) Fort Sill, OK 73503	1	Director Naval Research Laboratory ATTN: Code 2627 Washington, DC 20375
1	Director Night Vision Laboratory ATTN: DRSEL-NV-VI (A. Linz) Fort Belvoir, VA 22060	1	Department of the Air Force 1 HQ USAF/RDPS (Mr. Allan Eaffy) Washington, DC 20330
1	HQDA (DAMO-RQD) Washington, DC 20314	1	AFAPL/SFL Wright-Patterson AFB, OH 45433
1	Commander USAOCCS ATTN: ATSL-CD-CSN (LT Tate) Aberdeen Proving Ground, MD 21005		
2	Commander Field Command Defense Nuclear Agency ATTN: FCDNA/FCP (COL Hemler) FCDNA/FCPRX (MAJ Darda) Kirtland AFB, NM 87117		

No. Copies Addressee

Others

- | | |
|---|---|
| 1 | The BDM Corporation
ATTN: C. Somers
7915 Jones Branch Dr
McLean, VA 22101 |
| 1 | Science Applications, Inc
ATTN: W. Schilling
8400 Westpark Dr
McLean, VA 22101 |
| 1 | Standard Research Institute
ATTN: R. Baines
333 Ravenswood Ave
Menlo Park, CA 94025 |
| 1 | University of California
Lawrence Livermore Laboratories
D-Division (Mr. Blumenthal)
P.O. Box 808
Livermore, CA 94550 |
| 1 | University of California
Los Alamos Scientific Laboratories
ATTN: WPC-2 (Dr. Howard)
P.O. Box 1663
Los Alamos, NM 87544 |
| 1 | Sandia Laboratories
ATTN: Div-1311 (Mr. Lieber)
Kirtland AFB, NM 87117 |